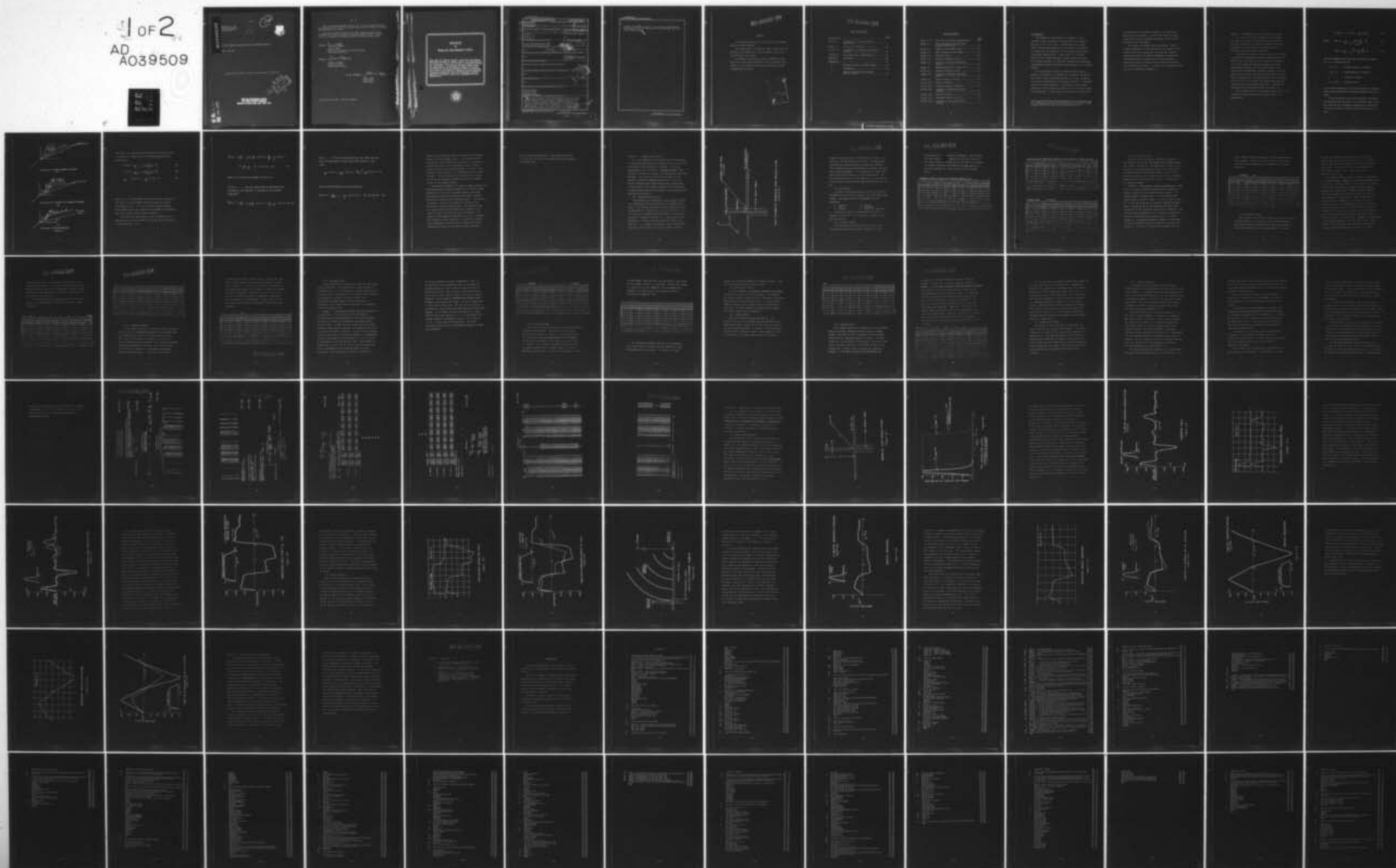
AD-A039 509 ROME AIR DEVELOPMENT CENTER GRIFFISS AFB N Y F/G 9/5
A TIME DOMAIN PROGRAM FOR WIRE ANTENNA ANALYSIS. (U)
APR 77 J POTENZA
UNCLASSIFIED RADC-TR-77-123 NL

1 OF 2
AD
A039509



ADA039509

12

RADC-TR-77-123
In-house Report
April 1977

A TIME DOMAIN PROGRAM FOR WIRE ANTENNA ANALYSIS

John Potenza

Approved for public release; distribution unlimited.

ROME AIR DEVELOPMENT CENTER
AIR FORCE SYSTEMS COMMAND
GRIFFISS AIR FORCE BASE, NEW YORK 13441

Some of the machine-produced data are not of the best quality, however, they are included for information purposes only and do not affect the technical understanding of this report.

This report has been reviewed by the RADC Information Office and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be releasable to the general public, including foreign nations.

APPROVED: Frank J Rehm

FRANK J. REHM
Asst Chief, Surveillance Technology Branch
Surveillance Division

APPROVED: Joseph L Ryerson

JOSEPH L. RYERSON
Technical Director
Surveillance Division

FOR THE COMMANDER: John P. Huss

JOHN P. HUSS
Acting Chief
Plans Office

Do not return this copy. Retain or destroy.

MISSION
of
Rome Air Development Center

RADC plans and conducts research, exploratory and advanced development programs in command, control, and communications (C^3) activities, and in the C^3 areas of information sciences and intelligence. The principal technical mission areas are communications, electromagnetic guidance and control, surveillance of ground and aerospace objects, intelligence data collection and handling, information system technology, ionospheric propagation, solid state sciences, microwave physics and electronic reliability, maintainability and compatibility.

AMERICAN REVOLUTION BICENTENNIAL
1776-1976

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER (14) RADC-TR-77-123	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER (9)	
4. TITLE (and Subtitle) (6) A TIME DOMAIN PROGRAM FOR WIRE ANTENNA ANALYSIS.	5. TYPE OF REPORT & PERIOD COVERED In-house - Final rept.
	6. PERFORMING ORG. REPORT NUMBER N/A
7. AUTHOR(s) (10) John Potenza	8. CONTRACT OR GRANT NUMBER(s) N/A
9. PERFORMING ORGANIZATION NAME AND ADDRESS Rome Air Development Center (OCTS) Griffiss Air Force Base, NY 13441	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS PE 62702F (16) 45060471 (17) Ø4
11. CONTROLLING OFFICE NAME AND ADDRESS Same (12) 142p.	12. REPORT DATE (11) Apr[redacted]77
	13. NUMBER OF PAGES 156
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Same	15. SECURITY CLASS. (of this report) UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Same

18. SUPPLEMENTARY NOTES

None

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Transient Analysis
Impulse Response
Computer Routines
Mathematical Modeling

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report discusses a user oriented computer routine to analyze antennas directly in the time domain. A number of sample cases were run and compared to previously published experimental results. A discussion of inputting data and output format is included to guide potential users. For the sake of completeness, a review of the time domain equations as well as the Fortran listing of the program, which were developed by MB Associates under Contracts RADC F30602-72X0008 over

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

309 050.

20.

and N00014-73-C-0099, are included. This information was published in a final report [1] and was distributed only to RADC and NRL, and is not available through DDC.

PREFACE

The analysis presented in this report was performed under Job Order 45060471.

The author wishes to thank Mr. Donald Harvey for his assistance and criticism in the reduction of data and preparation of the report.

In addition, the author wishes to acknowledge and thank Miss Jane Basenfelder/TIPN for the excellent reproduction of the measured data and the original art work contained in the report.

TABLE OF CONTENTS

LIST OF FIGURES

INTRODUCTION

The purpose of this report is to discuss a user-oriented computer routine* to analyze wire-type antennas directly in the time domain. Instructions for program input data and interpretation of the output are presented in concise form to permit use of the program with minimum theoretical understanding. Sample cases are presented which walk the reader through a typical problem solution. The computed results are critically compared with published experimental results to establish the validity of the computer routine.

The time domain approach is the most direct way to obtain the response of an antenna to a transient excitation. The output of the time domain solution is a time history of the current distribution on the antenna caused by a transient voltage or field excitation. Integration of this current provides the time varying far field or the receiving response of the antenna. By using Fourier transform the results can

*The computer routine was developed by MB Associates and has been implemented on computers at RADC, NRL, and NAVSEC under contract N00014-73-C-0099 and RADC F30602-72X0008.

be expressed in the frequency domain as the wide band response of the antenna. A single time domain solution yields data over the bandwidth associated with the time waveform of the excitation.

This report is divided into three parts. First a general description of the formulation is provided. Then program input and instructions for using the computer routines are given, including preparation of input data and interpretation of output for a specific example. Finally the output is discussed and critically compared with experimental results. A listing of the Fortran source program is included as Appendix A.

SECTION I Formulation of the Time Domain Equations

1.0 For the sake of completeness and providing a self-contained report, this section presents without rigorous derivation the integral equations which were developed and used in the Time Domain Program by MB Associates under government contract[1].

This computer routine is based upon the time domain equivalent of Packlington's Integral equation which is widely used for frequency domain numerical solutions. In the time domain the equation becomes an integro-differential equation and the unknown current is a function of two variables, position and time. The time domain equation can be derived by using Maxwell's time varying equations and the representation of the electric field in terms of the vector and scalar potentials and introduction of the retarded potential. Consider the general antenna structure depicted in Figure 1.1.

The electric field at an arbitrary point P $(\bar{r}, \theta, \phi)$ outside the volume current and charged source V' is expressed as

$$\bar{E}(\bar{r},t) = -\nabla\phi(\bar{r},t) - \frac{\partial}{\partial t}\bar{A}(\bar{r},t) \tag{1}$$

where
$$\bar{A}(\bar{r},t) = \frac{\mu}{4\pi}\int_{V'}\frac{J(\bar{r}',t-\frac{R}{C})}{R}\,dv' \tag{2}$$

$$\phi(\bar{r},t) = \frac{1}{4\pi\varepsilon}\int_{V'}\frac{\rho(\bar{r}',t-\frac{R}{C})}{R}\,dv' \tag{3}$$

are the retarded vector and scalar potentials at point $P(\bar{r}, \theta, \phi)$ respectively and

$J(\bar{r}', t')$ = current density in Volume V'

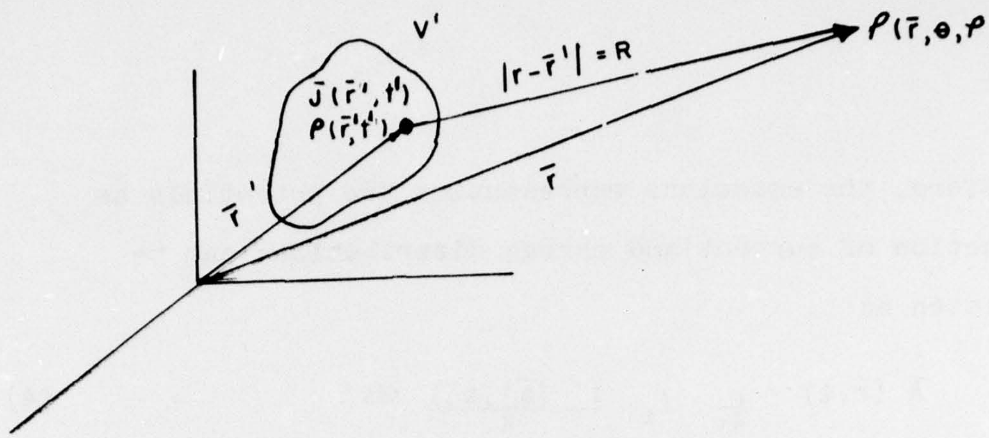
$\rho(\bar{r}', t')$ = charge density in volume V'

C = velocity of light

$t' = t-\frac{R}{C}$ = retarded time

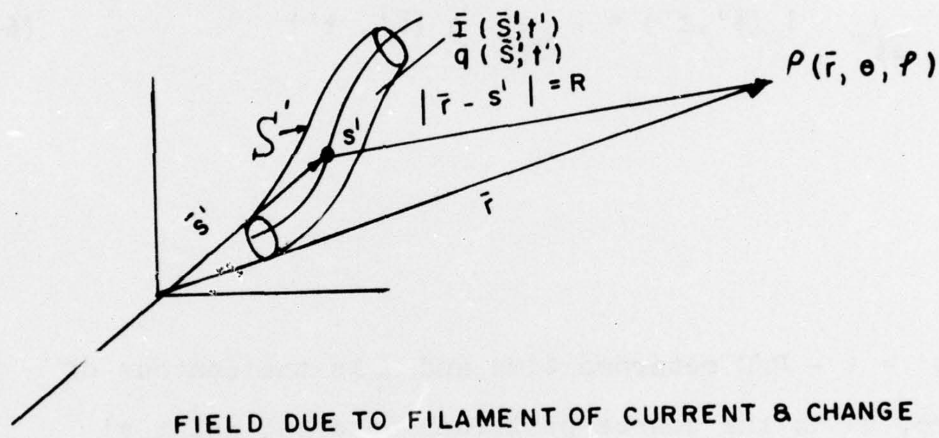
In the above expressions, the primed quantities represent source coordinates and the unprimed represent the field point.

If one considers a thin wire structure the current and charge are now confined to a filamentary path on the wire axis and the integration represented in equations (2) and (3) reduce to contour integrals along the wire axis.



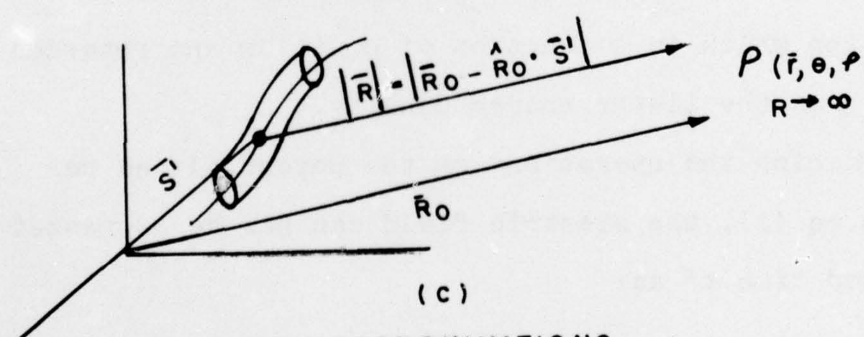
FIELD DUE TO VOLUME CURRENT & CHANGE

(a)



FIELD DUE TO FILAMENT OF CURRENT & CHANGE

(b)



(c)

FARFIELD APPROXIMATIONS

Figure 1.1

Therefore, the equations representing the potentials as a function of current and charge distributions can be rewritten as

$$\bar{A}\ (\bar{r},t) = \frac{\mu}{4\pi} \int_{\ell} \frac{I'\ (\bar{s}',t')}{R}\ ds' \tag{4}$$

$$\phi\ (\bar{r},t) = \frac{1}{4\pi\epsilon} \int_{\ell} \frac{q\ (\bar{s}',\ t')}{R}\ ds' \tag{5}$$

$$\frac{\partial}{\partial s'}\ I\ (\bar{s}',t') = -\frac{\partial}{\partial t'}\ q\ (\bar{s}',\ t') \tag{6}$$

where $t' = t - R/C$ retarded time and ℓ is the contour of the wire, $\bar{s}'$ is the source position vector at point s' (See Figure 1.1b). I $(\bar{s}',\ t')$ is the unknown current distribution which is a function of position and retarded time and q is the linear charge density.

Performing the operations on the potentials as required in eq (1), the electric field can now be expressed in retarded time t' as:

$$\overline{E}(\overline{r},t) = -\frac{\mu\hat{s}'}{4\pi}\int_{\ell}\Big[\frac{1}{R}\frac{\partial}{\partial t'}I(s',t') + \frac{C\overline{R}}{R^2}\frac{\partial}{\partial s'}I(s',t')$$

$$+C^2\frac{\overline{R}}{R^3}\frac{\partial}{\partial s'}\int_{o}^{t_1} I(s',t')dt'\Big]\,ds' \qquad (7)$$

where $\hat{s}'$ is unit vector tangent to wire at s'.

If now, $R \to \infty$ and only those terms are retained which decrease as 1/R, equation (7) reduces to the farfield expression

$$\overline{E}(\overline{r},t) = -\frac{\mu\hat{s}'}{4\pi}\int_{\ell}\Big[\frac{1}{R}\frac{\partial}{\partial t'}I(s',t') + C\frac{\overline{R}}{R^2}\frac{\partial}{\partial s'}I(s',t')\Big]\,ds' \quad (8)$$

Since $R \rightarrow \infty$ $\overline{R}$ can be replaced by $\overline{R}_o$, the vector from the point of observation to the origin (See Figure 1c), and using

$$\frac{\partial}{\partial s'} I(s',t') = \frac{d}{ds'} I(s',t') - \frac{\hat{R}_o \cdot \hat{s}'}{C} \frac{\partial}{\partial t'} I(s',t') \quad (9)$$

The far field formulation can be written as

$$\overline{E}_r(\overline{r},t) = - \frac{\mu l}{4\pi R_o} \int_{\ell'} \frac{\partial}{\partial t'} I(s',t') \left[\hat{S}' - (\hat{S}' \cdot R_o) \hat{R}_o \right] ds' \quad (10)$$

Equation 10 is the final form of the integro differential equation for the radiated field. It is obvious that if the current on the antenna is known both as a function of position and time, it is a relatively simple matter to compute the field or received response. The crux of the computer routine herein is the computation of the induced currents. While no attempt will be made in this report to develop the numerical techniques employed, a general outline of the concept follows.

The method of moments[1] is used to reduce equation (7) (with appropriate boundary conditions) to a set of linear equations to solve for the unknown current, involving both position and time rather than the single variable positions as occurs in the frequency domain. Subsectional collocation methods are applied to solve the linear equations for current coefficients. This requires that the antenna structure be divided into a number of segments and time be divided into equal increments. The details of the numerical techniques will not be presented here because of their complexity. However, the effects of the numerical techniques are to compute the current (position and time) at the center of each subsection

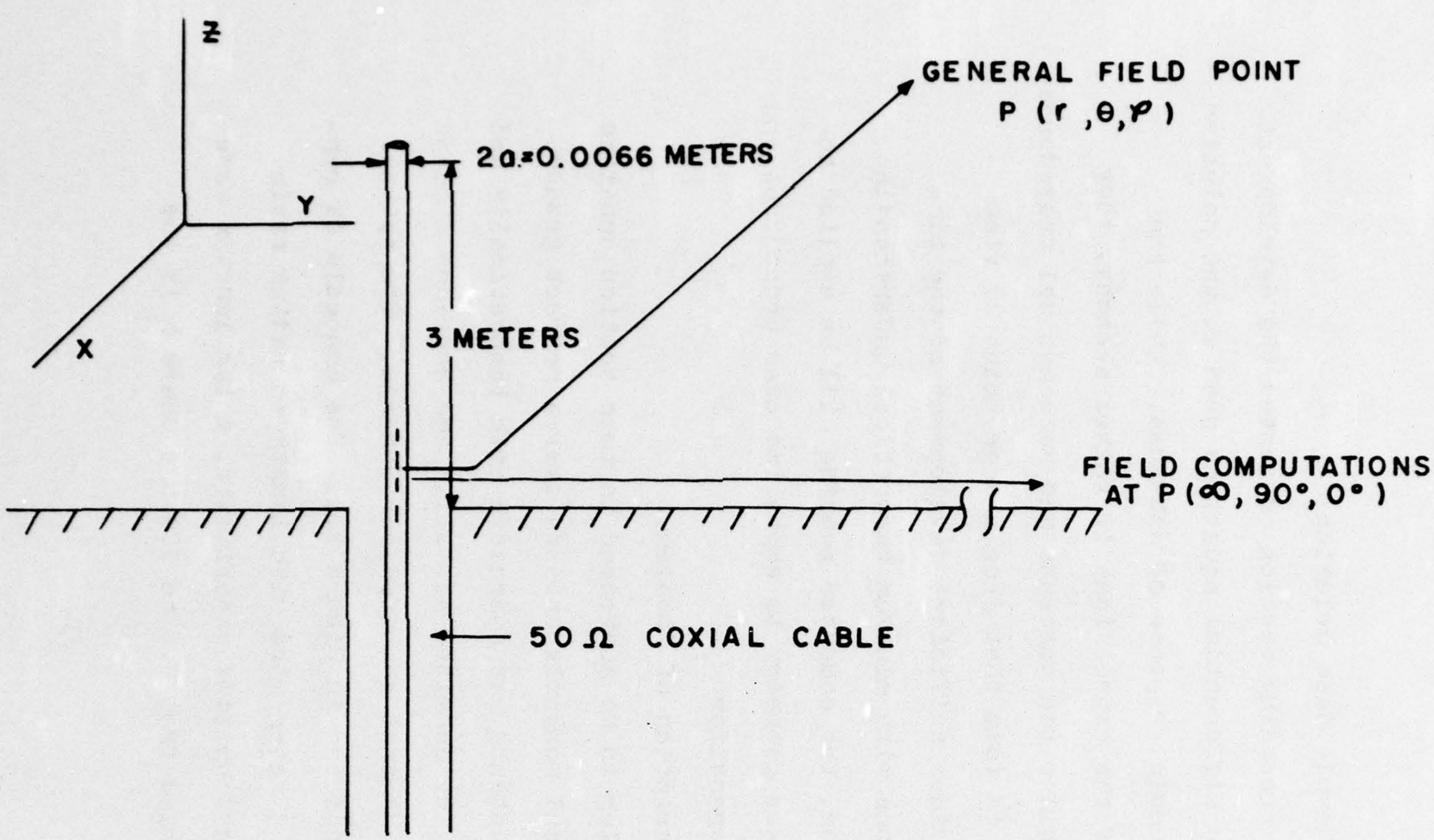
due to impressed excitation. Once these currents are computed the total field is determined through equations (8) through (10).

SECTION II Sample Case Solution

2.0 The preceding section presented the development of the integro differential equations used in the solution of the time domain response of antennas. While the development of the expressions is rather sketchy, they suffice to outline the concepts and mathematical procedures employed. It is felt that from a user point of view Section I provides sufficient development of the time domain equations with minimum theoretical understanding. In this section, the computer routine will be applied to a simple antenna geometry to emphasize user procedures and data input preparation.

2.1 Description of Problem.

The problem to be addressed in this section consists of a cylindrical monopole antenna over a perfect ground plane. This antenna was addressed both theoretically and experimentally by Schmitt, et. al.,[2] and provides a means of comparison against computed performance. The actual geometry is depicted in Figure 2.1. The monopole is represented by a 3 meter wire with length to radius ratio $h/a = 904$. In Schmitt's experiments, pulse lengths were used which varied from 0.2 to 2.0 h/c where h is the



THIN CYLINDRICAL MONOPOLE OVER GROUND PLANE

Figure 2.1

length of the wire and c is the velocity of light. The rise time for the pulses is 0.05 h/c. The excitations considered are voltage sources to obtain the radiated field response and field excitations to examine receiving time domain performance. The excitations are short base band time waveform voltage and field excitations. The time characteristics of the excitations are 3 db width of 0.16 h/c and 2.0 h/c with rise time on the order of 0.05 h/c.

2.2 Data Input.

In order to input this problem into the Time Domain (TIMDOM) program the problem must be cast into input data format. The input data can be separated into six categories.

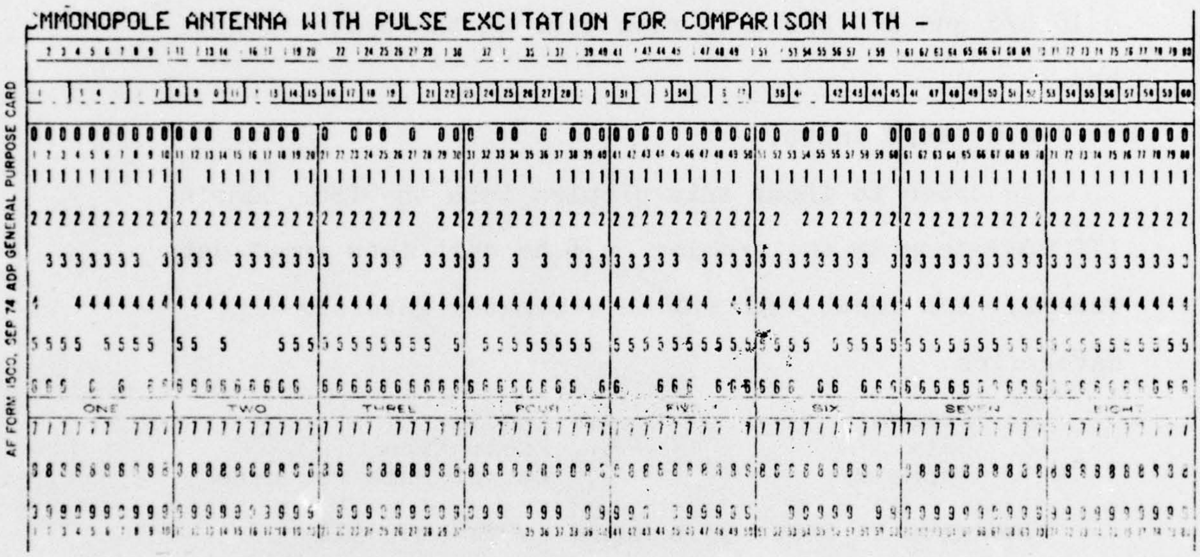
1. Comments	4. Loading
2. Data	5. Excitation
3. Geometry	6. Output Data requests

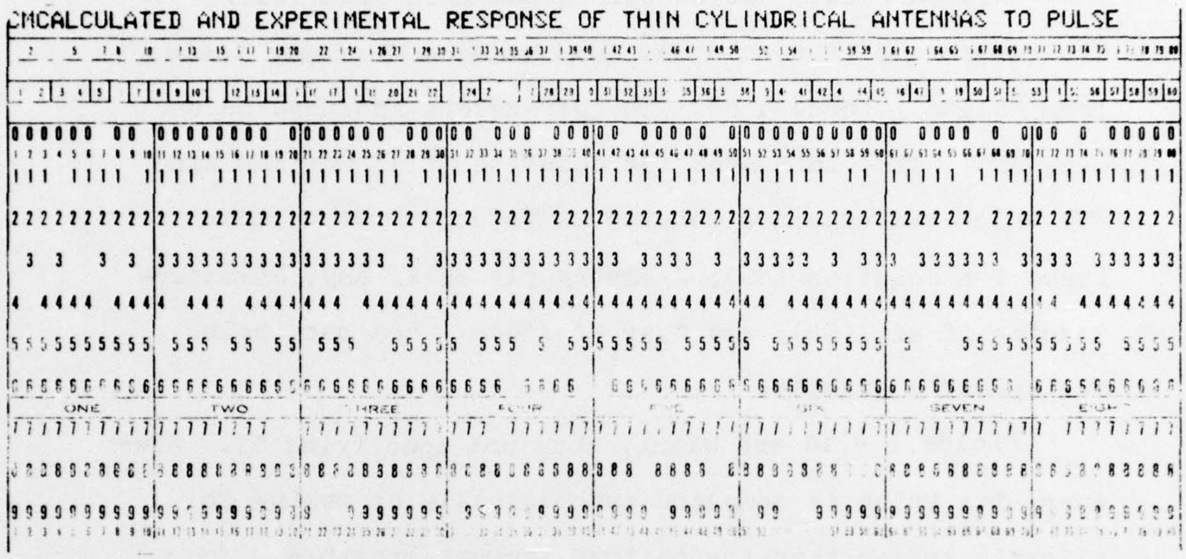
The data is input as punched cards and will be described in the following paragraphs as applied to the example problem.

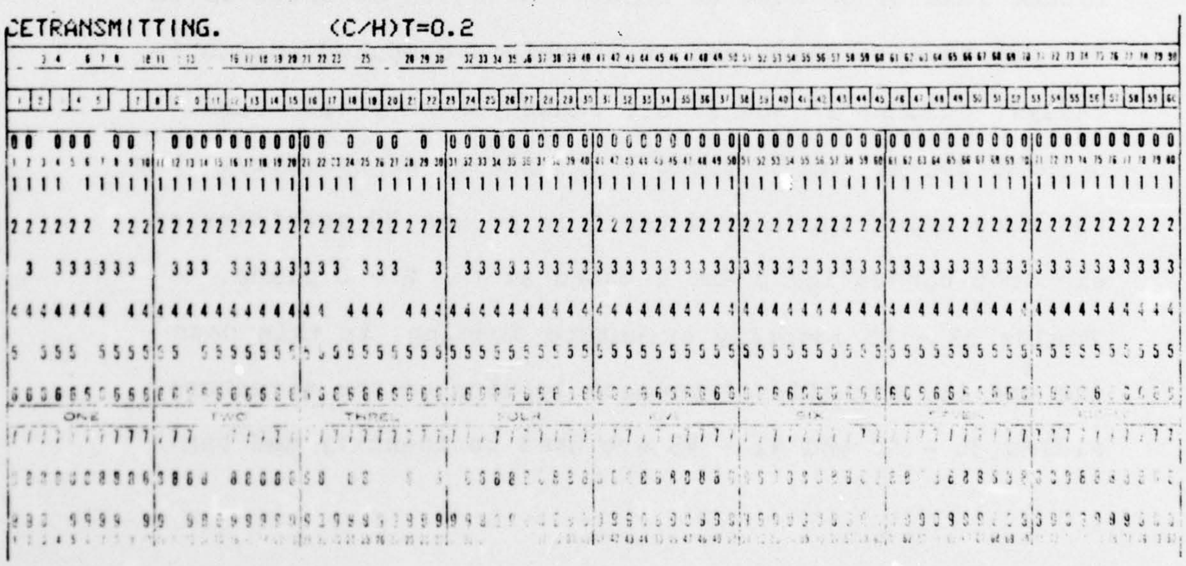
2.2.1 Comment Cards.

The data card deck must begin with one or more comment cards which may contain a description of the

problem or any other pertinent information. Each comment card must begin with the mnemonic CM and the contents of the card in columns 3 through 80 are printed in the output. All comment cards are read serially and <u>must end</u> with a card with mnemonic CE. This flags the end of comment cards.




CMCALCULATED AND EXPERIMENTAL RESPONSE OF THIN CYLINDRICAL ANTENNAS TO PULSE
AF FORM 1500, SEP 74 ADP GENERAL PURPOSE CARD

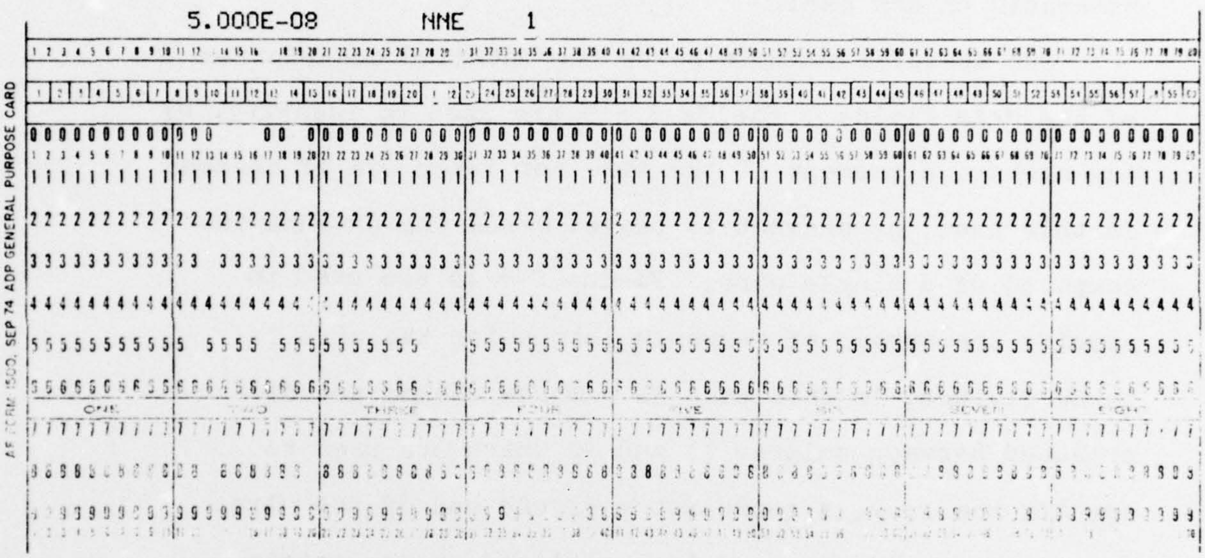

CETRANSMITTING.   (C/H)T=0.2
AF FORM 1500, SEP 74 ADP GENERAL PURPOSE CARD

2.2.2 First Data Cards.

This card does not contain a mnemonic identifier. The contents of the card are divided into 7 usable fields from column 1 through 45. The parameters included on the card are: Time increment for solution (TI); Maximum time for which currents are computed (TMAX); Number of time steps for solution (NS); Symmetry planes if any; Structure loading if any (SL); and type of loads. See card below for specific example.

Fields 1 - 10 are blank, thus not specifying TI. However, its value is computed automatically by making it slightly larger than the maximum segment length/c. The integer number occupying field 11 - 20 specifies TMAX. Either TMAX or NS must be input. NS which occupies field 21 - 25 is left blank and is therefore computed automatically. Columns 26 and 27 are unused on this data card. Columns 28, 29 and 30 specify the planes of symmetry. For the case being considered the E in column 30 specifies an electric conducting plane located at the Z = 0 plane. Fields 31 - 35 identify structure loading, in this case the one (1) signifies impedance loading on the structure. Fields 36 - 40 and 41 - 45 are used to identify and tag

those segments of the wire antenna for which currents are to be printed. In this case where they are left blank, currents will be printed for all segments as a default condition.



2.2.3 Geometry Cards.

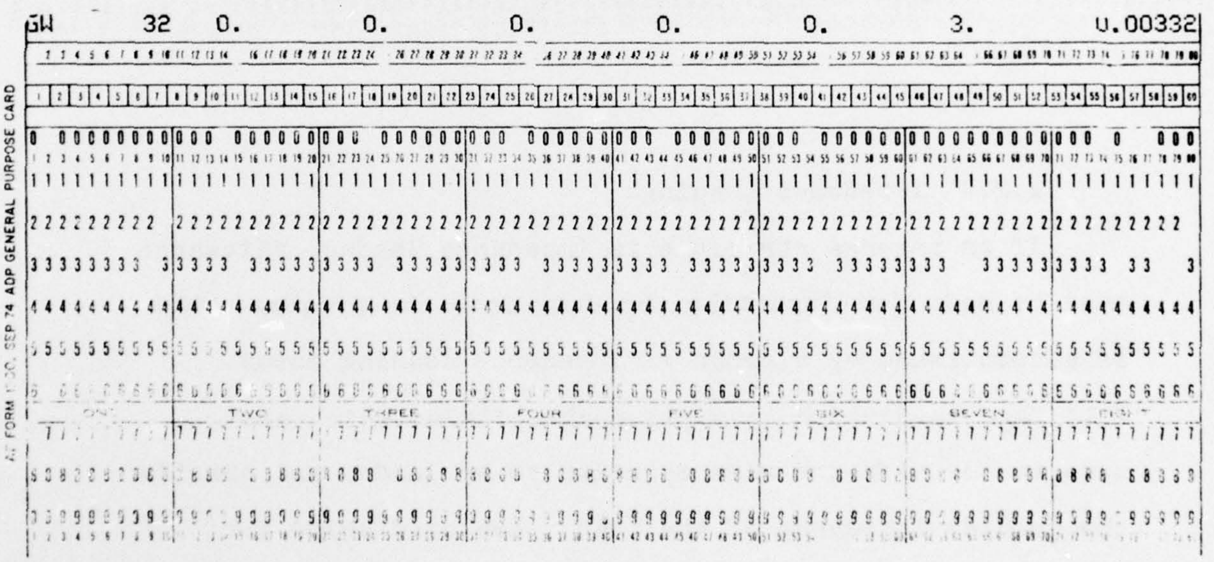
The purpose of the geometry cards is to input the antenna structure geometry. Specifically these cards (mnemonic) GW generate lengths of straight wire between two points in space and divide them into an equal number of segments.

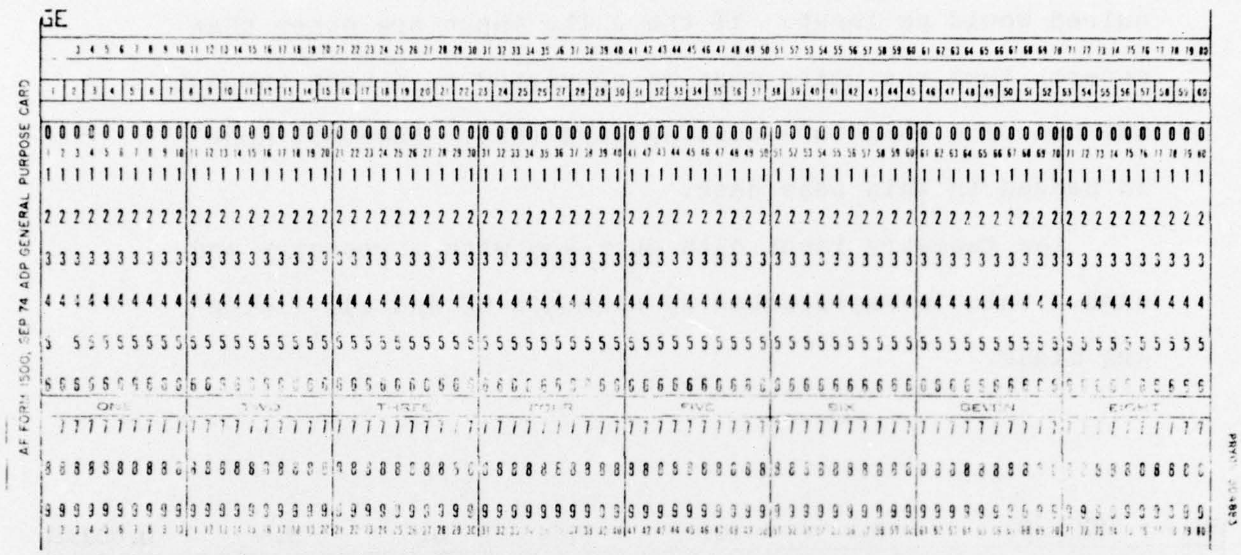
The card is divided into 10 fields which are used to identify the end points of the wire (in x, y, z space), identify the number of segments, and provide the radius of the wire. Any number of wires can be generated in this fashion, however, only a single wire can be generated on one card.

See specific example below for detailed description of the data fields. Fields 3 - 5 are used to identify, by tag number, a segment of the wire for later use if needed. In this case, no segment is tagged since the antenna is composed of a single wire. Fields 6 - 10 are used to specify the number of segments into which the wire is to be divided, in this case 32. Three fields are distributed between columns 11 and 40 which are used to specify the x, y, z coordinates of one end of the wire. In this case the wire end is positioned at the origin. Columns 41 - 70 are used to specify the other end of the wire. The data in column 64 places the wire 3 meters along the Z axis. See Figure 2.1. Fields 71 - 80 are used to specify the radius of the wire in this case 0.00332 meters. Since this case only involves a single wire antenna only one Geometry card is used. If a more

complex structure is being modeled, as many cards as required would be input. If the units input are other than meters, then the units must be converted to meters through the use of a structure scaling card (GS). However, none is needed in this case deck.

The geometry input data must end with a geometry end card. This is represented by mnemonic GE and all fields are blank.

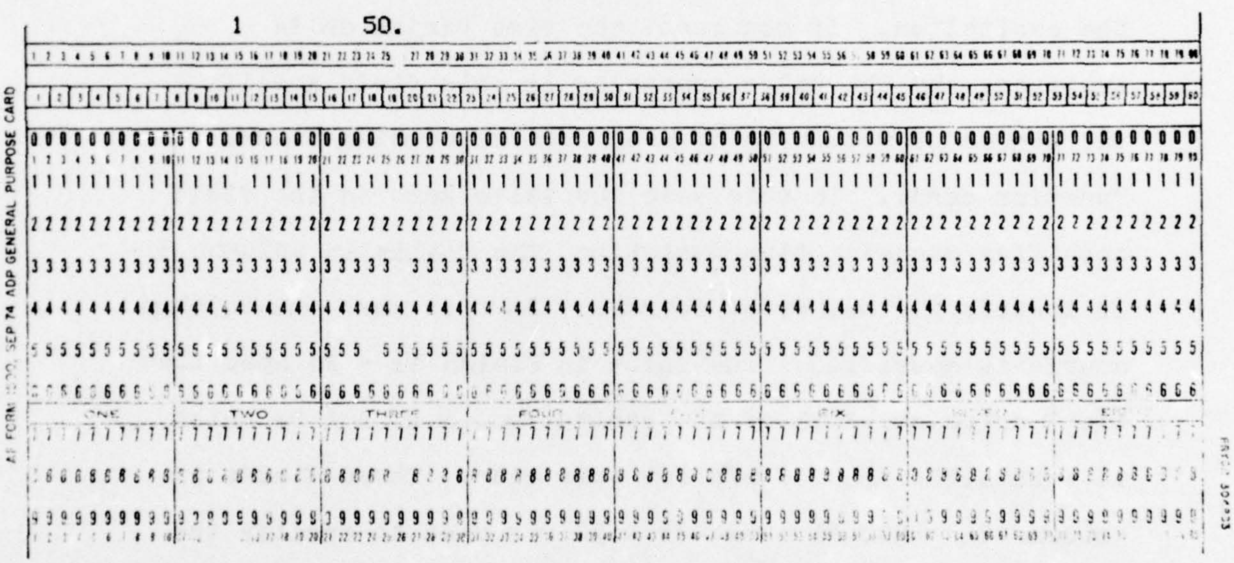




2.2.4 Impedance Loading.

If an antenna structure is impedance loaded, allowance must be made for inputting the data into the program. This is accomplished by a group of impedance loading cards. This card is divided into six usable fields which are used to identify the wire segments to be loaded and specify the complex impedance of the load. See card below for format details. Fields 1 - 5 are used to specify the number of impedance cards. If non zero, the program

searches for succeeding loading cards. In this case that field is zero implying this is the last impedance loading card. Fields 6 - 10 and 11 - 15 are used to identify, by tag number, the segment to be loaded. In this case segment one (1) is identified. Fields 21 - 30, 31 - 40, 41 - 50 are used to specify in ohms the resistance, inductance, and capacitance of the impedance. In this case, a 50 ohm pure resistance is input to simulate the coax feed point impedance.

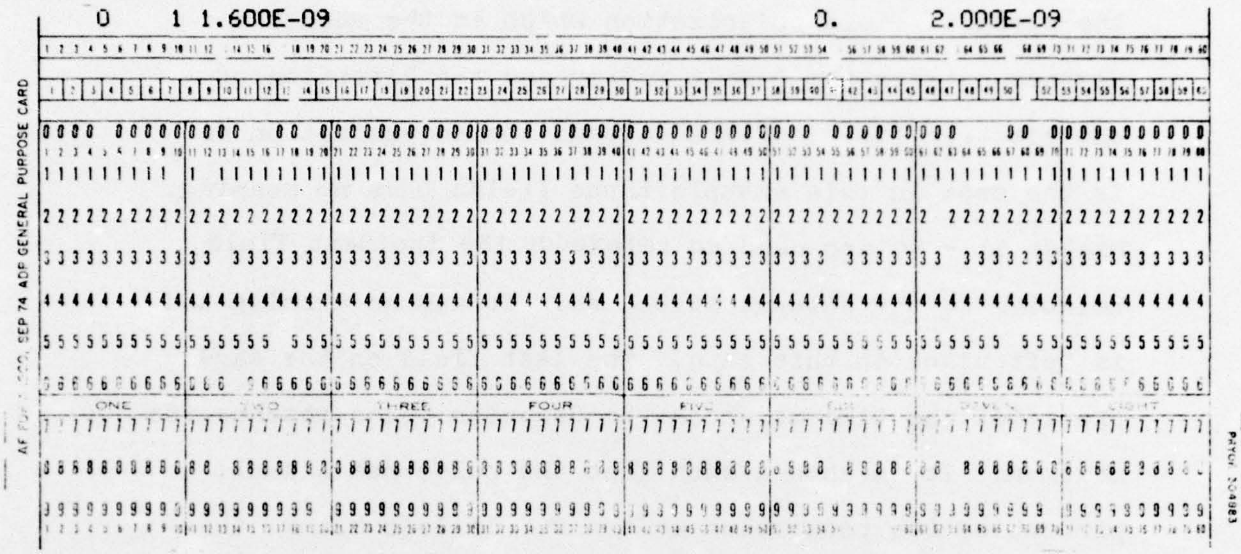


2.2.5 Excitation Card.

Having described the geometry and loads, the excitation must be specified. This information is input through the excitation card. It is divided into 8 fields which are used to specify type of excitation, i.e. voltage or field excitation; number of sources; 3 db pulse width of the excitation; incidence angle for field excitation. A discussion of the various fields references the excitation card of the example case.

Fields 1 - 5 are used to identify the time variation of the excitation. If non zero, the time variation is arbitrary and the value appearing in this field specifies the number of values to be read from succeeding Source Function cards. In this case the value zero in the field specifies Gaussian time variation. The fields in columns 5 - 10 specify the number of sources. In this case one voltage source is specified. The value in Fields 11 - 20 specifies the 3 db pulse width of the excitation, i.e. 1.6 ns which is equivalent to 0.16 h/c for this case. Three fields in columns 21 - 30, 31 - 40, 41 - 50 are used to specify the θ, $\emptyset$, η parameters of the incidence angle for field excitation. The field source position is specified by

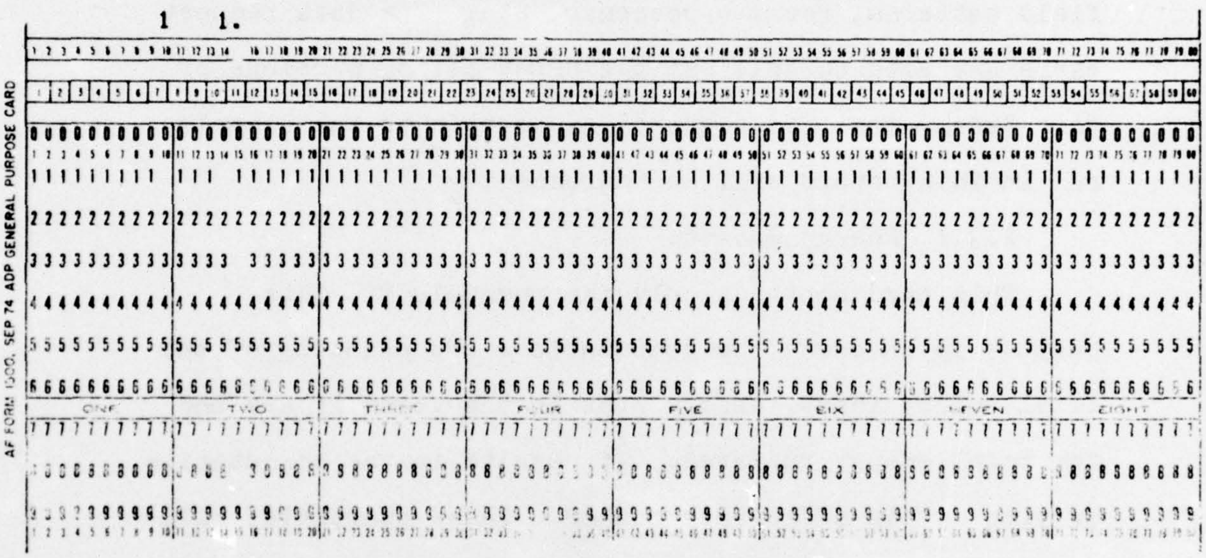
Θ, and $\emptyset$ (convention spherical coordinates), and η is the electric field polarization which is the angle in degrees between the Θ unit vector and the direction of the electric field. For a voltage source excitation as is the case in this example these fields have no meaning. Fields 51 - 60 are used to reference the incident field relative to the antenna structure. It has no meaning and is left blank in this case. The last field on the card specifies the time at which the Gaussian pulse reaches its maximum. It is chosen such that the pulse has a small percent of its peak value at t = 0. If left blank, the program computes an appropriate value ≈ 3% of peak at t = 0). In this case it was specified. This card completely specifies all the parameters required to input the excitation.



2.2.6 Source Voltage.

This card is used to specify the peak voltages of the individual sources used. It consists of three usable fields which identify the segments on which sources will be located, and input the peak voltage of the sources. On this card (see below) of the input data deck, fields 1 - 5 and 6 - 10 are used to specify the tag number of the source segment and the segment number of a set of segments, respectively. In this case since there is only

one tag number (only one wire) the tag field is left blank and the segment number 1 is specified. That is, the source is located at the base segment of the 32 segment wire monopole. The peak value of the source is unity as specified in Fields 11 - 20.



The preceding paragraphs describe the preparation of a sample input data deck which was prepared to input the problem into the computer. The next set of data

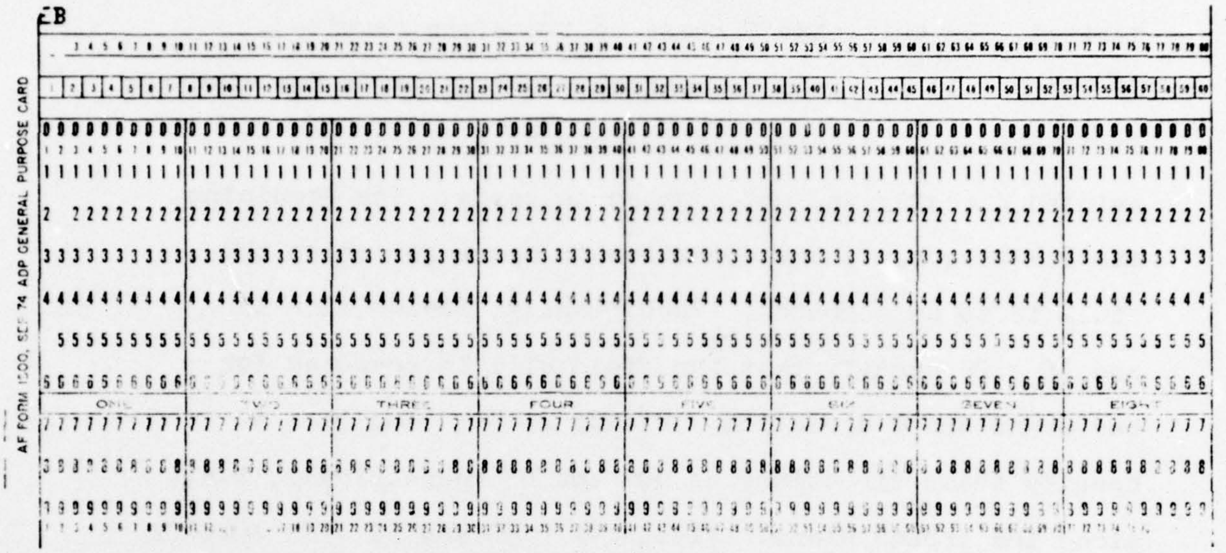
cards to be discussed pertains to requesting output. This set is referred to as Data Request Cards.

2.3 Data Request Cards.

A set of commands is required to identify and extract the proper data from the output solution. For example, the program computes induced current, impedance, admittance, field patterns, receive response, etc. The data request cards are used to call out and print all or portions of this data. Specific requests as pertain to the example run in this report will be addressed.

2.3.1 Energy Budget.

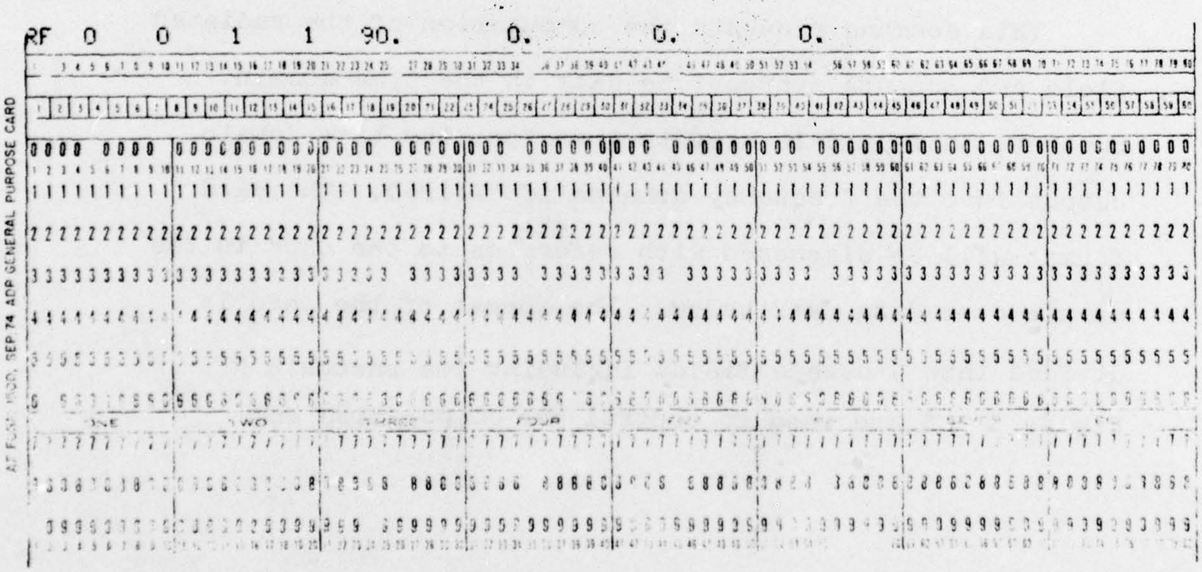
This card contains only the mnemonic EB. Its purpose is to cause the computations and printing of the total energy input, energy lost in the loads if any and the total energy radiated. It results in the computation of time domain efficiency number. These quantities will be discussed in the output data section of the report.



2.3.2 Radiated Field.

This command requests the computation of the radiated field and outputs appropriate data in the time domain. Further, this card is used to transform the time domain output into the frequency domains if desired. The card format will be discussed with reference to the card in the sample case data deck below. The format of the card is divided into 9 usable fields including the mnemonic RF. Fields 3 - 5 are used to control the computation and

printing of the Fourier Transform and punch cards if desired. In this case 0 means no transform required. Fields 6 - 10 are used to control punching cards for electric field output. In this case the electric field will be printed directly and not punched on cards. The remaining six fields specify the direction in space for which the field is to be computed. Specifically, fields 11 - 15 and 16 - 20 specify that the field will be computed for one θ and one $\emptyset$ direction. Fields 21 - 30 and 31 - 40 specify these directions as 90° and 0° respectively, i.e. along the ground plane. Fields 41 - 50 and 51 - 60 specify the θ and $\emptyset$ stepping increment for field computation. In this case both are zero since only one direction is of interest.



2.4 The preceding paragraphs describe the preparation of a data input deck for a specific case. Recall the problem was to determine the radiated field of a monopole antenna on a perfect conducting ground plane fed by a 50 ohm coaxial cable (see Figure 2.1). The sample input data deck was presented to input the problem parameters, i.e. geometry, excitation, loading, etc. and request current computation and far field transmitting response. The following paragraphs will present and discuss the output of the sample case. The radiated and received responses will be presented and compared to experimental results in Section III.

2.5 Program Data Output.

The purpose of this portion is to familarize the reader with the raw output of this program. The output presented is a result of the input problem discussed in paragraph 2.1. Although the output of the program is automatically identified by labels, each portion of the sample output has been identified by circled numbers for easy reference by the reader. The sample output which is discussed below is located at the end of this section.

2.5.1 Output Discussion.

The program title Time Domain Antenna Modeling Program appears at 1 automatically with each output as the first heading. The comment cards which were presented as input data follow the heading at 2 and can contain as many statements as desired. At 3 the first data card is read and printed. This card contains various time stepping information as well as any problem symmetry (see input data card discussion paragraph 2.2.2).

Next the geometry input cards are printed at 4. In this case, a single monopole is modeled by 32 segments with wire radius 0.0032 meters. These outputs identify the wires by tag numbers. The parameters specified by the geometry cards are used to determine the segment centers, lengths, and orientation (α, β orientation angles) which are printed in the output at 5. This print-out also contains segment interconnection data and tag numbers. The I- I+ notation is used to reference current flow, i.e. current flows from negative to positive. The wire information is summarized at 6.

The table automatically printed out at 7 summarizes the time stepping information used in the solution. It

provides the number of time steps and time stepping interval. If any one of these quantities was not input as data (see first data input card paragraph 2.2.2), it is automatically computed and printed here.

Any options such as symmetry planes, impedance loading, etc. are printed at location 8. This location also identifies the segment for which current will be printed.

The parameters of the excitation are provided at 9. In addition to printing the input excitation data card, the Gaussian characteristic time information is provided. As pointed out in this example, the source has 11.5% of its peak value at t = 0. This value was specified as data input otherwise it would have been automatically computed and printed here.

At location 10, all voltage source information is summarized. It specifies the number of sources, location and peak voltages. If field excitation had been used this section would be replaced by angle of incidence information.

At this point, the program begins the solution for the currents on the segments. The results are printed

out as a block of data at 11. The currents are printed from left to right in order of increasing segment number for each time step. In this case there are 160 time steps of ≈ 0.3 ns each. The value of the source strength at each time step is the final entry of each block, labeled excitation.

The results of the Energy Budget request card are printed at 12. The Time Domain Efficiency is defined as the ratio of the total energy radiated to the total energy input and accounts for any energy lost in loads on the structure. If no loads were present in the problem this efficiency would be 100%.

The next block of data 13 in the output is the result of the request for radiated field computations. It provides the electric field (volts/meter) variation for each time step at a specified point in space (theta and phi). It also provides polarization tilt information which is also a useful performance parameter.

The preceding paragraphs have presented a walk through discussion of the use of the time domain program to input a problem and interpret the output. The program was written to emphasize user-ease with a minimum amount

of previous training in numerical techniques or computer understanding. In the next section the results of the sample problem will be analyzed and compared with experimental results.

```
*****************************************
TIME DOMAIN ANTENNA MODELING PROGRAM                                   1
*****************************************

MONOPOLE ANTENNA WITH PULSE EXCITATION FOR COMPARISON WITH -

CALCULATED AND EXPERIMENTAL RESPONSE OF THIN CYLINDRICAL ANTENNAS TO PULSE   2
EXCITATION,   BY H.J. SCHMITT, C.W. HARRISON, JR., AND C.S.WILLIAMS, JR.
IEEE TRANS. ANT. PROP.,   MARCH 1966.

TRANSMITTING.        (C/H)T=0.2

*** FIRST DATA CARD** 0.        0.50000E-07   0  NNE   1   0   0          3
```

- - - STRUCTURE SPECIFICATION - - -

COORDINATES MUST BE INPUT IN
METERS OR BE SCALED TO METERS
BEFORE STRUCTURE INPUT IS ENDED

WIRE NO.	X1	Y1	Z1	X2	Y2	Z2	RADIUS	NO. OF SEG.	FIRST SEG.	LAST SEG.	TAG NO.
1	0.	0.	0.	0.	0.	3.00000	0.00332	32	1	32	0

4

- - - - SEGMENTATION DATA - - - -   5

COORDINATES IN METERS

I+ AND I- INDICATE THE SEGMENTS BEFORE AND AFTER I

SEG. NO.	COORDINATES OF SEG. CENTER X	Y	Z	SEG. LENGTH	ORIENTATION ANGLES ALPHA	BETA	WIRE RADIUS	CONNECTION DATA I-	I	I+	TAG NO.
1	0.	0.	0.04688	0.09375	90.00000	0.	0.00332	20001	1	2	0
2	0.	0.	0.14063	0.09375	90.00000	0.	0.00332	1	2	3	0
3	0.	0.	0.23438	0.09375	90.00000	0.	0.00332	2	3	4	0
4	0.	0.	0.32813	0.09375	90.00000	0.	0.00332	3	4	5	0
5	0.	0.	0.42188	0.09375	90.00000	0.	0.00332	4	5	6	0
6	0.	0.	0.51562	0.09375	90.00000	0.	0.00332	5	6	7	0
7	0.	0.	0.60938	0.09375	90.00000	0.	0.00332	6	7	8	0
8	0.	0.	0.70312	0.09375	90.00000	0.	0.00332	7	8	9	0
9	0.	0.	0.79688	0.09375	90.00000	0.	0.00332	8	9	10	0
10	0.	0.	0.89063	0.09375	90.00000	0.	0.00332	9	10	11	0
11	0.	0.	0.98437	0.09375	90.00000	0.	0.00332	10	11	12	0
12	0.	0.	1.07813	0.09375	90.00000	0.	0.00332	11	12	13	0
13	0.	0.	1.17188	0.09375	90.00000	0.	0.00332	12	13	14	0
14	0.	0.	1.26563	0.09375	90.00000	0.	0.00332	13	14	15	0
15	0.	0.	1.35938	0.09375	90.00000	0.	0.00332	14	15	16	0
16	0.	0.	1.45312	0.09375	90.00000	0.	0.00332	15	16	17	0

17	0.	0.	1.54688	0.09375	90.00000	0.	0.00332	16	17	18	0
18	0.	0.	1.64062	0.09375	90.00000	0.	0.00332	17	18	19	0
19	0.	0.	1.73438	0.09375	90.00000	0.	0.00332	18	19	20	0
20	0.	0.	1.82812	0.09375	90.00000	0.	0.00332	19	20	21	0
21	0.	0.	1.92188	0.09375	90.00000	0.	0.00332	20	21	22	0
22	0.	0.	2.01563	0.09375	90.00000	0.	0.00332	21	22	23	0
23	0.	0.	2.10938	0.09375	90.00000	0.	0.00332	22	23	24	0
24	0.	0.	2.20313	0.09375	90.00000	0.	0.00332	23	24	25	0
25	0.	0.	2.29688	0.09375	90.00000	0.	0.00332	24	25	26	0
26	0.	0.	2.39063	0.09375	90.00000	0.	0.00332	25	26	27	0
27	0.	0.	2.48438	0.09375	90.00000	0.	0.00332	26	27	28	0
28	0.	0.	2.57812	0.09375	90.00000	0.	0.00332	27	28	29	0
29	0.	0.	2.67188	0.09375	90.00000	0.	0.00332	28	29	30	0
30	0.	0.	2.76562	0.09375	90.00000	0.	0.00332	29	30	31	0
31	0.	0.	2.85937	0.09375	90.00000	0.	0.00332	30	31	32	0
32	0.	0.	2.95312	0.09375	90.00000	0.	0.00332	31	32	0	0

NUMBER OF SEGMENTS= 32
TOTAL WIRE LENGTH= 3.00000 M.
MAXIMUM SEGMENT LENGTH= 0.09375 M. 6

****

TIME STEPPING INTERVAL= 0.31300E-09 SEC. 7
MAXIMUM TIME= 0.49767E-07 SEC.
NUMBER OF TIME STEPS= 160

****

OPTIONS SELECTED- - -
THE Z=0 PLANE IS AN ELECTRIC SYMMETRY PLANE
MODEL HAS IMPEDANCE LOADING 8
CURRENTS WILL BE PRINTED FOR SEGMENTS 1 THROUGH 32

- - - - STRUCTURE LOADING - - - -

TAG NO.	INCREMENT	SEG. NO.	RESISTANCE (OHMS)	INDUCTANCE (HENRYS)	CAPACITANCE (FARADS)
0	1	1	0.5000E 02	0.	

*** EXCITATION DATA CARD** 0 1 0.16000E-08 0. 0. 0. 0. 0.20000E-08

- - - - EXCITATION - - - -

TIME DEPENDENCE- GAUSSIAN, PEAK AT TIME=TMAX - - EXP(-(A**2)*(TIME-TMAX)**2) 9

A= 0.73589E 09, TMAX= 0.20000E-08

SOURCE HAS 11.462 PERCENT OF PE[illegible]

```
VOLTAGE SOURCES -

SOURCE   SOURCE SEGMENT        PEAK VOLTAGE
 NO.      TAG  INC,  SEG,         (VOLTS)
          NO.         NO,
  1        0     1     1          1.0000
```

(10)

```
LENGTH OF COEFFICIENT ARRAY=      24243
        4 BLOCKS OUTPUT TO FILE 11.         2 BLOCKS OUTPUT TO FILE 12
TIME FOR INITIALIZATION OF CONSTANTS=  485.000 SEC.
SOLUTION WILL USE CURRENTS FROM CORE STORAGE ONLY
LENGTH OF CURRENT OUTPUT BLOCK IS    2304 WORDS.
NUMBER OF TIME STEPS USED DURING SOLUTION=       65

                                   - - - - CURRENT SOLUTION - - - -
STEP      TIME       SEGMENT CURRENTS IN AMPS (READ ACROSS) - - - - - -
 NO.      (SEC.)     LAST NUMBER IN EACH BLOCK REPRESENTS SOURCE STRENGTH
```

(11)

```
  1 0.              0.286E-03  0.329E-04  0.449E-05  0.595E-06  0.791E-07  0.105E-07  0.140E-08  0.186E-09  0.248E-10  0.329E-11
                    0.438E-12  0.583E-13  0.775E-14  0.103E-14  0.137E-15  0.182E-16  0.242E-17  0.322E-18  0.429E-19  0.570E-20
                    0.759E-21  0.101E-21  0.134E-22  0.178E-23  0.237E-24  0.316E-25  0.420E-26  0.558E-27  0.743E-28  0.988E-29
                    0.132E-29  0.186E-30
                    EXCITATION     0.1146E 00

  2 0.31300E-09     0.671E-03  0.208E-03  0.465E-04  0.867E-05  0.148E-05  0.241E-06  0.379E-07  0.582E-08  0.878E-09  0.131E-09
                    0.192E-10  0.280E-11  0.404E-12  0.581E-13  0.829E-14  0.118E-14  0.167E-15  0.236E-16  0.331E-17  0.464E-18
                    0.649E-19  0.905E-20  0.126E-20  0.175E-21  0.243E-22  0.336E-23  0.465E-24  0.641E-25  0.884E-26  0.122E-26
                    0.167E-27  0.238E-28
                    EXCITATION     0.2141E 00

  3 0.62600E-09     0.112E-02  0.555E-03  0.199E-03  0.545E-04  0.123E-04  0.250E-05  0.472E-06  0.845E-07  0.146E-07  0.244E-08
                    0.398E-09  0.638E-10  0.101E-10  0.157E-11  0.241E-12  0.367E-13  0.555E-14  0.832E-15  0.124E-15  0.183E-16
                    0.27 E-17  0.396E-18  0.577E-19  0.838E-20  0.121E-20  0.175E-21  0.251E-22  0.360E-23  0.515E-24  0.735E-25
                    0.105E-25  0.155E-26
                    EXCITATION     0.3597E 00
```

•
•
•
•
•
•

```
•
•
•
          0.190E-04  0.698E-05
          EXCITATION  0.

157 0.48828E-07  0.775E-04  0.915E-04  0.119E-03  0.144E-03  0.150E-03  0.131E-03  0.921E-04  0.471E-04  0.148E-04  0.117E-04
                 0.466E-04  0.119E-03  0.218E-03  0.331E-03  0.442E-03  0.537E-03  0.606E-03  0.642E-03  0.644E-03  0.616E-03
                 0.562E-03  0.492E-03  0.415E-03  0.338E-03  0.266E-03  0.204E-03  0.151E-03  0.108E-03  0.747E-04  0.486E-04
                 0.280E-04  0.104E-04
                 EXCITATION  0.

158 0.49141E-07  0.539E-04  0.745E-04  0.109E-03  0.141E-03  0.156E-03  0.148E-03  0.117E-03  0.737E-04  0.325E-04  0.832E-05
                 0.125E-04  0.503E-04  0.121E-03  0.217E-03  0.327E-03  0.438E-03  0.534E-03  0.604E-03  0.641E-03  0.643E-03
                 0.613E-03  0.559E-03  0.489E-03  0.412E-03  0.336E-03  0.265E-03  0.202E-03  0.148E-03  0.104E-03  0.693E-04
                 0.405E-04  0.152E-04
                 EXCITATION  0.

159 0.49454E-07  0.512E-04  0.689E-04  0.955E-04  0.122E-03  0.140E-03  0.144E-03  0.130E-03  0.103E-03  0.666E-04  0.322E-04
                 0.110E-04  0.139E-04  0.485E-04  0.116E-03  0.212E-03  0.325E-03  0.437E-03  0.534E-03  0.604E-03  0.639E-03
                 0.640E-03  0.609E-03  0.556E-03  0.487E-03  0.410E-03  0.334E-03  0.261E-03  0.197E-03  0.143E-03  0.966E-04
                 0.573E-04  0.216E-04
                 EXCITATION  0.

160 0.49767E-07  0.632E-04  0.715E-04  0.822E-04  0.955E-04  0.110E-03  0.124E-03  0.130E-03  0.124E-03  0.102E-03  0.690E-04
                 0.331E-04  0.828E-05  0.857E-05  0.432E-04  0.113E-03  0.212E-03  0.329E-03  0.437E-03  0.533E-03  0.601E-03
                 0.636E-03  0.637E-03  0.607E-03  0.553E-03  0.484E-03  0.407E-03  0.329E-03  0.256E-03  0.190E-03  0.131E-03
                 0.788E-04  0.299E-04
                 EXCITATION  0.
TIME FOR CURRENT SOLUTION=-1825.000 SEC.

RUNNING TIME=-1340.000 SEC.

*** DATA CARD**EB  0    0    0    0 0.          0.          0.          0.


                    - - - ENERGY BUDGET - - -                                    <—— (12)

                  SOURCES -

                     SOURCE  SEGMENT   ENERGY INPUT
                       NO.     NO.      (JOULES)
                        1       1       0.4881E-11

                  LOADS -
                     SEGMENT   RESISTANCE  ENERGY LOSS
                       NO.       (OHMS)      (JOULES)
                        1      0.5000E 02  0.1787E-11

                       TOTAL ENERGY INPUT = 0.4881E-11 JOULES
                       ENERGY LOST IN LOADS = 0.1787E-11 JOULES
                       TOTAL ENERGY RADIATED = 0.3094E-11 JOULES
                       TIME DOMAIN EFFICIENCY =  63.39 PERCENT

RUNNING TIME=-1329.000 SEC.

*** DATA CARD**RF  0    0    1    1 0.90000E 02 0.          0.          0.
```

- - - - RADIATED FIELD - - - -

THETA= 90.000
PHI = 0.

STEP NO.	TIME (SEC.)	ELECTRIC FIELD (V/M) THETA	ELECTRIC FIELD (V/M) PHI	POL. (DEG.)
1	-0.31300E-09	0.9718E-02	0.	0.
2	0.	0.2805E-01	0.	0.
3	0.31300E-09	0.4855E-01	0.	0.
4	0.62600E-09	0.7540E-01	0.	0.
5	0.93900E-09	0.1058E 00	0.	0.
6	0.12520E-08	0.1334E 00	0.	0.
7	0.15650E-08	0.1509E 00	0.	0.
8	0.18780E-08	0.1527E 00	0.	0.
9	0.21910E-08	0.1373E 00	0.	0.
10	0.25040E-08	0.1084E 00	0.	0.
11	0.28170E-08	0.7343E-01	0.	0.
12	0.31300E-08	0.4007E-01	0.	0.
13	0.34430E-08	0.1376E-01	0.	0.
14	0.37560E-08	-0.3551E-02	0.	180.000
15	0.40690E-08	-0.1277E-01	0.	180.000
16	0.43820E-08	-0.1622E-01	0.	180.000
17	0.46950E-08	-0.1638E-01	0.	180.000
18	0.50080E-08	-0.1508E-01	0.	180.000
19	0.53210E-08	-0.1339E-01	0.	180.000
20	0.56340E-08	-0.1180E-01	0.	180.000
21	0.59470E-08	-0.1045E-01	0.	180.000
22	0.62600E-08	-0.9345E-02	0.	180.000
23	0.65730E-08	-0.8448E-02	0.	180.000
24	0.68860E-08	-0.7708E-02	0.	180.000
25	0.71990E-08	-0.7092E-02	0.	180.000
26	0.75120E-08	-0.6591E-02	0.	180.000
27	0.78250E-08	-0.6219E-02	0.	180.000
28	0.81380E-08	-0.6053E-02	0.	180.000
29	0.84510E-08	-0.6281E-02	0.	180.000
30	0.87640E-08	-0.7299E-02	0.	180.000
31	0.90770E-08	-0.9834E-02	0.	180.000
32	0.93900E-08	-0.1506E-01	0.	180.000
33	0.97030E-08	-0.2462E-01	0.	180.000
34	0.10016E-07	-0.4039E-01	0.	180.000
35	0.10329E-07	-0.6388E-01	0.	180.000
36	0.10642E-07	-0.9547E-01	0.	180.000
37	0.10955E-07	-0.1335E 00	0.	180.000
38	0.11268E-07	-0.1739E 00	0.	180.000
39	0.11581E-07	-0.2104E 00	0.	180.000
40	0.11894E-07	-0.2356E 00	0.	180.000
41	0.12207E-07	-0.2432E 00	0.	180.000
42	0.12520E-07	-0.2297E 00	0.	180.000
43	0.12833E-07	-0.1957E 00	0.	180.000
44	0.13146E-07	-0.1470E 00	0.	180.000
45	0.13459E-07	-0.9225E-01	0.	180.000
46	0.13772E-07	-0.4159E-01	0.	180.000
47	0.14085E-07	-0.3141E-02	0.	180.000
48	0.14398E-07	0.1901E-01	0.	0.
49	0.14711E-07	0.2571E-01	0.	0.
50	0.15024E-07	0.2177E-01	0.	0.

STEP NO.	TIME (SEC.)	ELECTRIC FIELD (V/M) THETA	ELECTRIC FIELD (V/M) PHI	POL. (DEG.)
81	0.24727E-07	-0.3706E-02	0.	180.000
82	0.25040E-07	-0.3845E-02	0.	180.000
83	0.25353E-07	-0.2718E-02	0.	180.000
84	0.25666E-07	-0.1039E-02	0.	180.000
85	0.25979E-07	0.4977E-03	0.	0.
86	0.26292E-07	0.1431E-02	0.	0.
87	0.26605E-07	0.1640E-02	0.	0.
88	0.26918E-07	0.1311E-02	0.	0.
89	0.27231E-07	0.8030E-03	0.	0.
90	0.27544E-07	0.4713E-03	0.	0.
91	0.27857E-07	0.5387E-03	0.	0.
92	0.28170E-07	0.1070E-02	0.	0.
93	0.28483E-07	0.2048E-02	0.	0.
94	0.28796E-07	0.3493E-02	0.	0.
95	0.29109E-07	0.5576E-02	0.	0.
96	0.29422E-07	0.8655E-02	0.	0.
97	0.29735E-07	0.1323E-01	0.	0.
98	0.30048E-07	0.1985E-01	0.	0.
99	0.30361E-07	0.2894E-01	0.	0.
100	0.30674E-07	0.4074E-01	0.	0.
101	0.30987E-07	0.5508E-01	0.	0.
102	0.31300E-07	0.7136E-01	0.	0.
103	0.31613E-07	0.8840E-01	0.	0.
104	0.31926E-07	0.1045E 00	0.	0.
105	0.32239E-07	0.1175E 00	0.	0.
106	0.32552E-07	0.1251E 00	0.	0.
107	0.32865E-07	0.1255E 00	0.	0.
108	0.33178E-07	0.1177E 00	0.	0.
109	0.33491E-07	0.1020E 00	0.	0.
110	0.33804E-07	0.7997E-01	0.	0.
111	0.34117E-07	0.5473E-01	0.	0.
112	0.34430E-07	0.2992E-01	0.	0.
113	0.34743E-07	0.9106E-02	0.	0.
114	0.35056E-07	-0.5082E-02	0.	180.000
115	0.35369E-07	-0.1158E-01	0.	180.000
116	0.35682E-07	-0.1108E-01	0.	180.000
117	0.35995E-07	-0.5785E-02	0.	180.000
118	0.36308E-07	0.1373E-02	0.	0.
119	0.36621E-07	0.7593E-02	0.	0.
120	0.36934E-07	0.1100E-01	0.	0.
121	0.37247E-07	0.1107E-01	0.	0.
122	0.37560E-07	0.8514E-02	0.	0.
123	0.37873E-07	0.4819E-02	0.	0.
124	0.38186E-07	0.1524E-02	0.	0.
125	0.38499E-07	-0.3670E-03	0.	180.000
126	0.38812E-07	-0.6659E-03	0.	180.000
127	0.39125E-07	0.9255E-04	0.	0.
128	0.39438E-07	0.1026E-02	0.	0.
129	0.39751E-07	0.1350E-02	0.	0.
130	0.40064E-07	0.6890E-03	0.	0.

13

52	0.15650E-07	0.6660E-02	0.	0.
53	0.15963E-07	0.3446E-02	0.	0.
54	0.16276E-07	0.3655E-02	0.	0.
55	0.16589E-07	0.5311E-02	0.	0.
56	0.16902E-07	0.6508E-02	0.	0.
57	0.17215E-07	0.6486E-02	0.	0.
58	0.17528E-07	0.5639E-02	0.	0.
59	0.17841E-07	0.4779E-02	0.	0.
60	0.18154E-07	0.4409E-02	0.	0.
61	0.18467E-07	0.4510E-02	0.	0.
62	0.18780E-07	0.4816E-02	0.	0.
63	0.19093E-07	0.5185E-02	0.	0.
64	0.19406E-07	0.5734E-02	0.	0.
65	0.19719E-07	0.6706E-02	0.	0.
66	0.20032E-07	0.8291E-02	0.	0.
67	0.20345E-07	0.1054E-01	0.	0.
68	0.20658E-07	0.1339E-01	0.	0.
69	0.20971E-07	0.1670E-01	0.	0.
70	0.21284E-07	0.2018E-01	0.	0.
71	0.21597E-07	0.2340E-01	0.	0.
72	0.21910E-07	0.2580E-01	0.	0.
73	0.22223E-07	0.2681E-01	0.	0.
74	0.22536E-07	0.2598E-01	0.	0.
75	0.22849E-07	0.2319E-01	0.	0.
76	0.23162E-07	0.1866E-01	0.	0.
77	0.23475E-07	0.1302E-01	0.	0.
78	0.23788E-07	0.7129E-02	0.	0.
79	0.24101E-07	0.1931E-02	0.	0.
80	0.24414E-07	-0.1809E-02	0.	180.000

132	0.40690E-07	-0.2897E-02	0.	180.000
133	0.41003E-07	-0.4966E-02	0.	180.000
134	0.41316E-07	-0.6778E-02	0.	180.000
135	0.41629E-07	-0.8266E-02	0.	180.000
136	0.41942E-07	-0.9497E-02	0.	180.000
137	0.42255E-07	-0.1052E-01	0.	180.000
138	0.42568E-07	-0.1126E-01	0.	180.000
139	0.42881E-07	-0.1150E-01	0.	180.000
140	0.43194E-07	-0.1103E-01	0.	180.000
141	0.43507E-07	-0.9703E-02	0.	180.000
142	0.43820E-07	-0.7599E-02	0.	180.000
143	0.44133E-07	-0.4990E-02	0.	180.000
144	0.44446E-07	-0.2273E-02	0.	180.000
145	0.44759E-07	0.1406E-03	0.	0.
146	0.45072E-07	0.1933E-02	0.	0.
147	0.45385E-07	0.2931E-02	0.	0.
148	0.45698E-07	0.3128E-02	0.	0.
149	0.46011E-07	0.2658E-02	0.	0.
150	0.46324E-07	0.1763E-02	0.	0.
151	0.46637E-07	0.7325E-03	0.	0.
152	0.46950E-07	-0.1571E-03	0.	180.000
153	0.47263E-07	-0.7175E-03	0.	180.000
154	0.47576E-07	-0.8972E-03	0.	180.000
155	0.47889E-07	-0.7945E-03	0.	180.000
156	0.48202E-07	-0.6232E-03	0.	180.000
157	0.48515E-07	-0.6453E-03	0.	180.000
158	0.48828E-07	-0.1099E-02	0.	180.000
159	0.49141E-07	-0.2152E-02	0.	180.000
160	0.49454E-07	-0.3899E-02	0.	180.000

TOTAL ENERGY RADIATED = 0.4[illegible]33E-12 JOULES/STERADIAN (THETA= 90,000 DEG., PHI= 0. DEG.)
TIME DOMAIN POWER GAIN = -10.[illegible] DB
TIME DOMAIN DIRECTIVE GAIN = -[illegible],02 DB

RUNNING TIME=-1304.000 SEC.

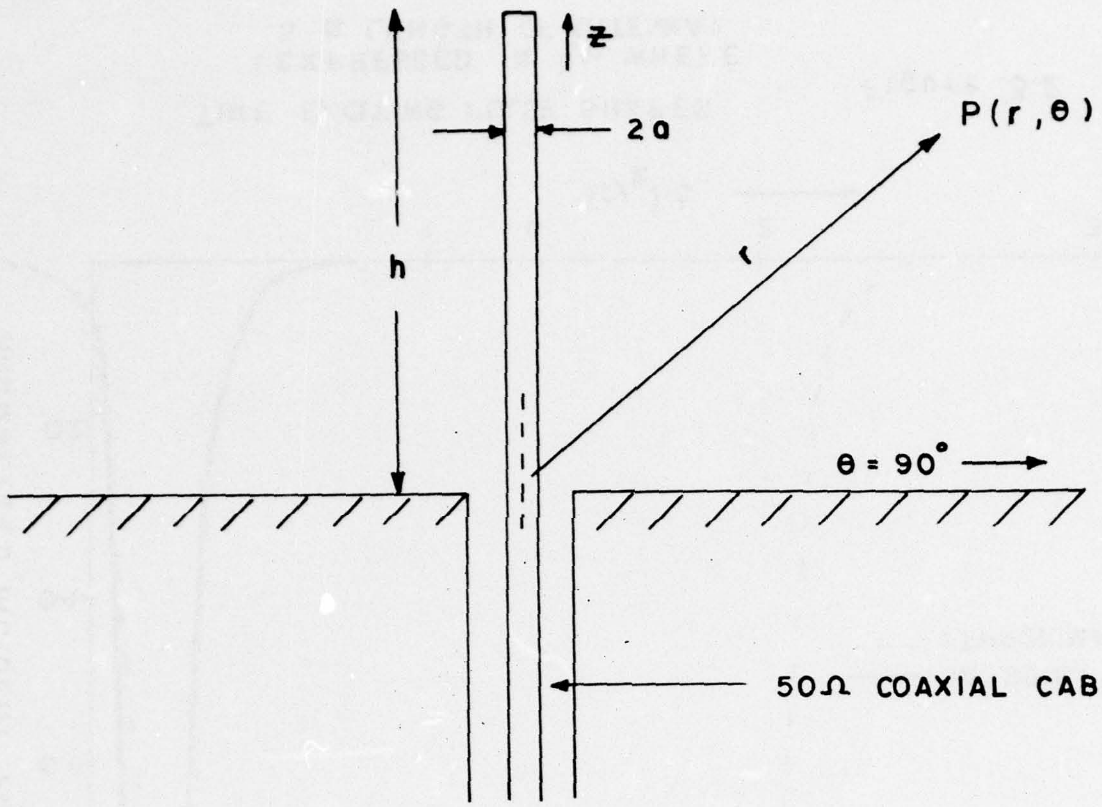
*** DATA CARD** [illegible] 0 0 [illegible] 0 0. 0. 0. 0.

SECTION III Comparison of Computed and Measured Results

3.0 The purpose of this section is to present and analyze the response of a monopole antenna to short base-band pulses. Both the transmitting and receiving computed responses will be presented and compared to measurements previously published.[2] The computed results were obtained by plotting the output of the TIMDOM program previously discussed.

3.1 Transmitting Response.

The cylindrical monopole and its zero resistivity ground plane and coaxial feed geometry are shown in Figure 3.1. The specific antenna geometry and excitation used in the computation was scaled in the time domain to the actual measurement geometry.[2] Two dc pulse excitations were used and are shown in Figure 3.2. The excitations are plotted with respect to h/c, the one-way travel time on the antenna where "h" is the monopole length and "c" is the free space velocity of propagation. The narrow excitation is a Gaussian pulse with 3db pulse width of 0.16 h/c (1.6 ns) since h is three meters in the computed case. The broader excitation represents a rectangular pulse with 3db width of 2.0 h/c.



MONOPOLE OVER GROUND PLANE

Figure 3.1

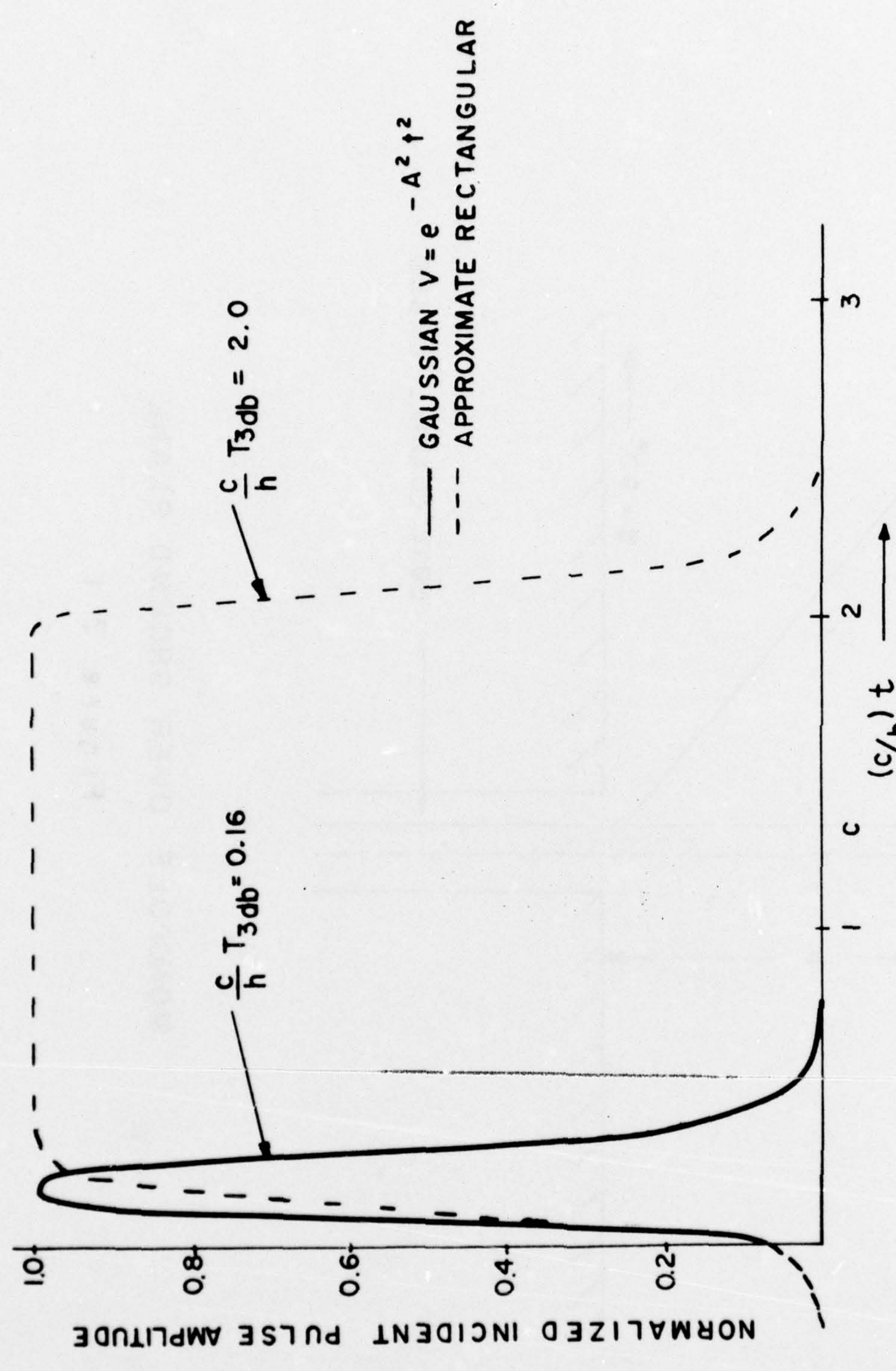
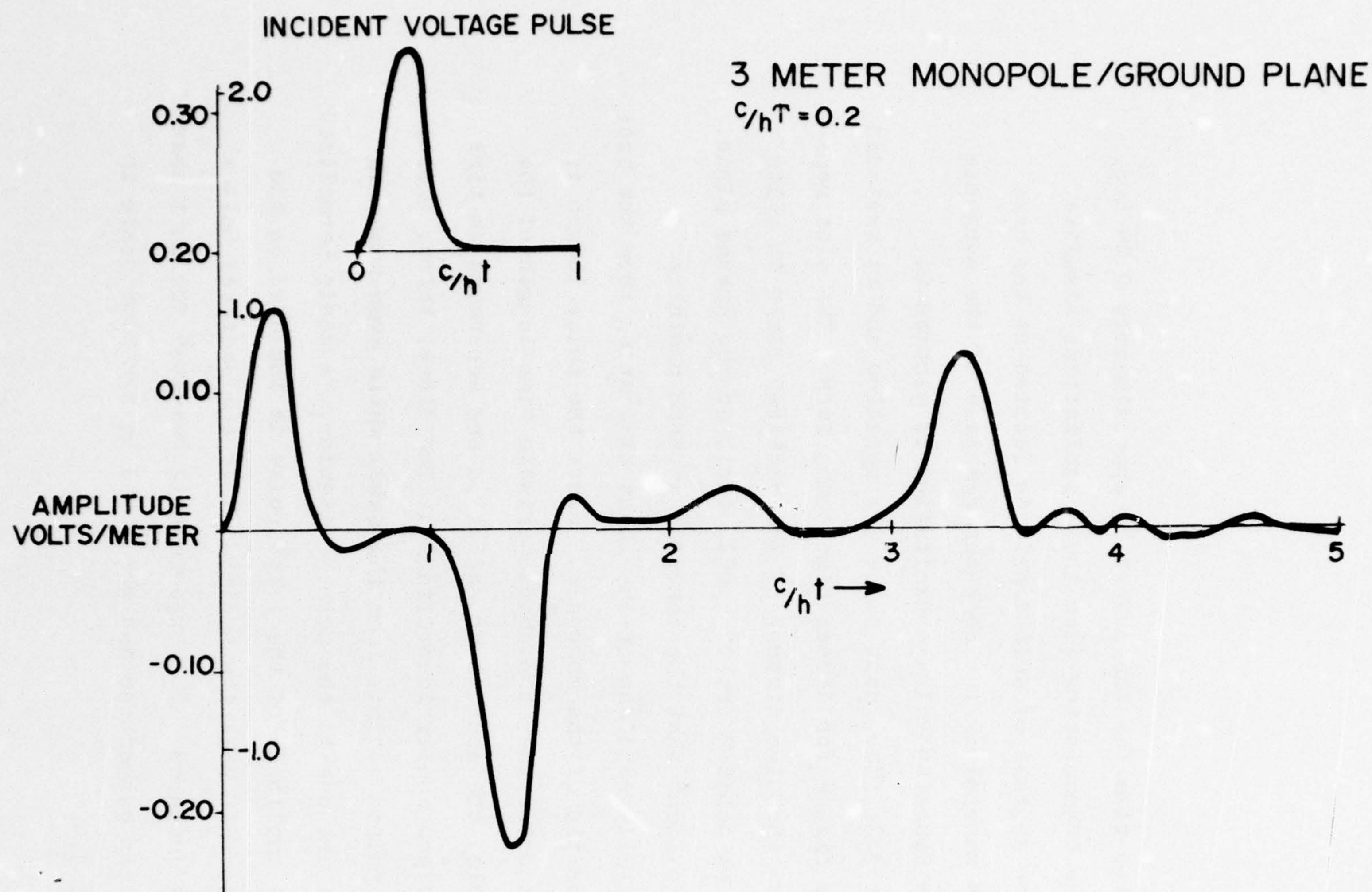


Figure 3.2

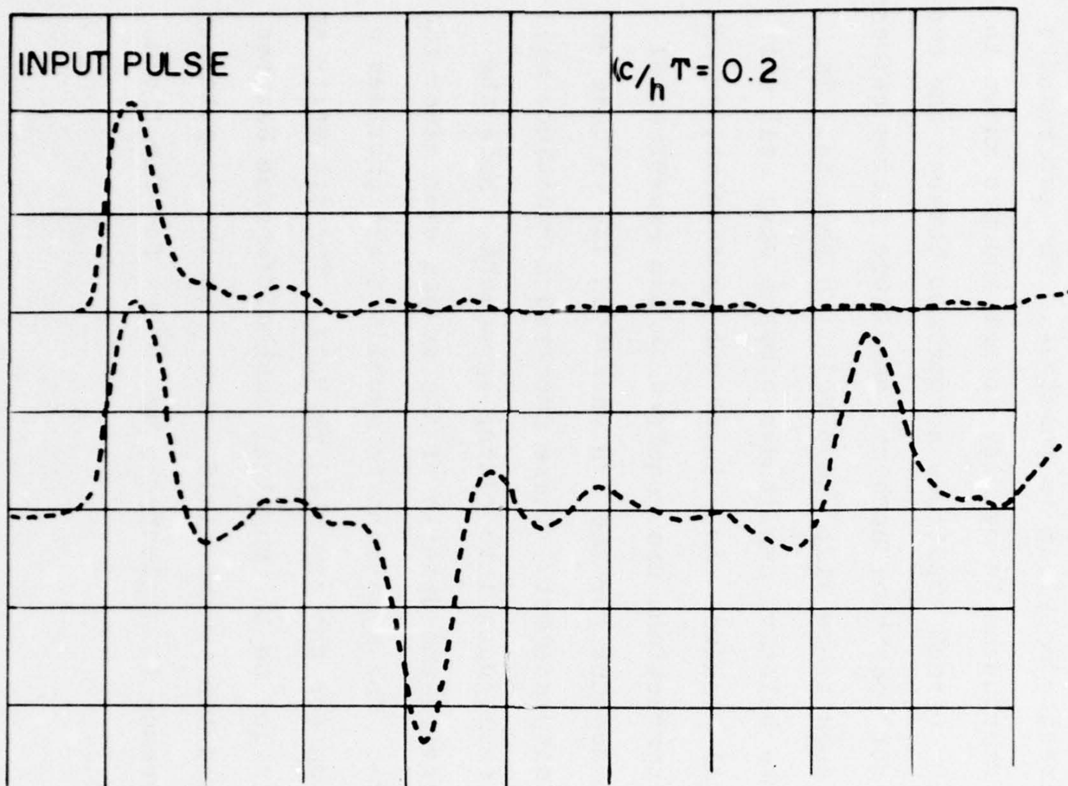
TIME EXCITING PULSE SHAPES
(EXPRESSED IN $h/_c$ WHERE
h IS LENGTH OF ANTENNA)

The rise time for the pulses is approximately 0.05 h/c. For the transmitting case these excitations simulate voltage sources of unit amplitude located at the base of the monopole. The computed response of the monopole to the Gaussian voltage excitation is plotted in Figure 3.3. The input pulse was rescaled and is included in the figure for direct time comparison. The plot represents the time history of the radiated field in volts/meter at point P (r, 90°, 0°), i.e., at the ground plane. It is evident that the monopole antenna radiates essentially replicas of the input excitation from the base and the tip of the monopole. Since the pulse length is much shorter than the characteristic time-length of the monopole, the sources of radiation are separated in time by h/c and easily identifiable. The first, third, fifth, etc., pulses radiate from the base, while even-numbered pulses are due to end point radiation. A scale normalized to the amplitude of the first pulse is included on the plot so that relative amplitudes of the radiated pulses can be compared. The experimental results[2] for this case have been reproduced and replotted in h/c time scale in Figure 3.4.



RADIATION FIELD

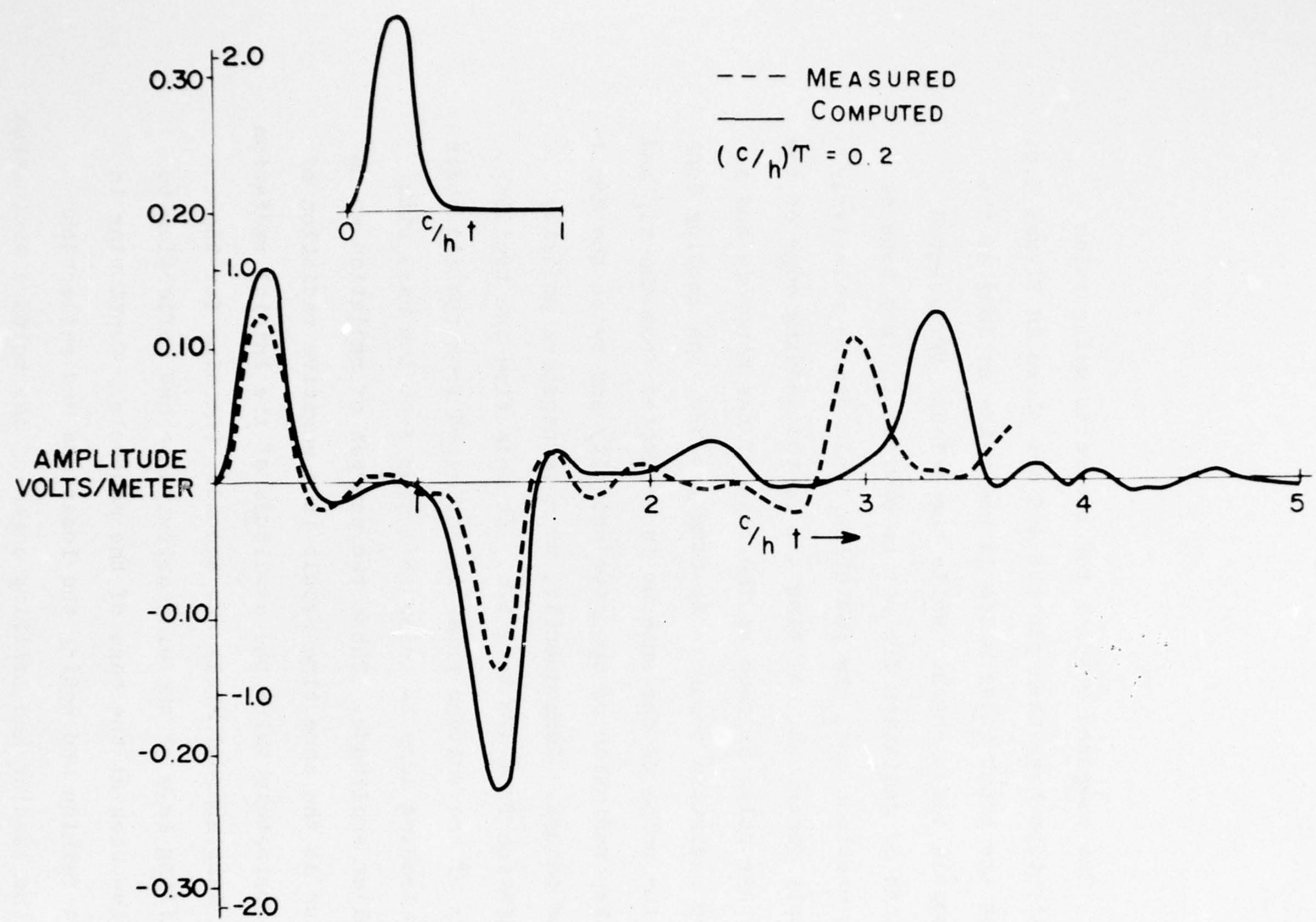
Figure 3.3



MEASURED RADIATION FIELD

FIGURE 3.4

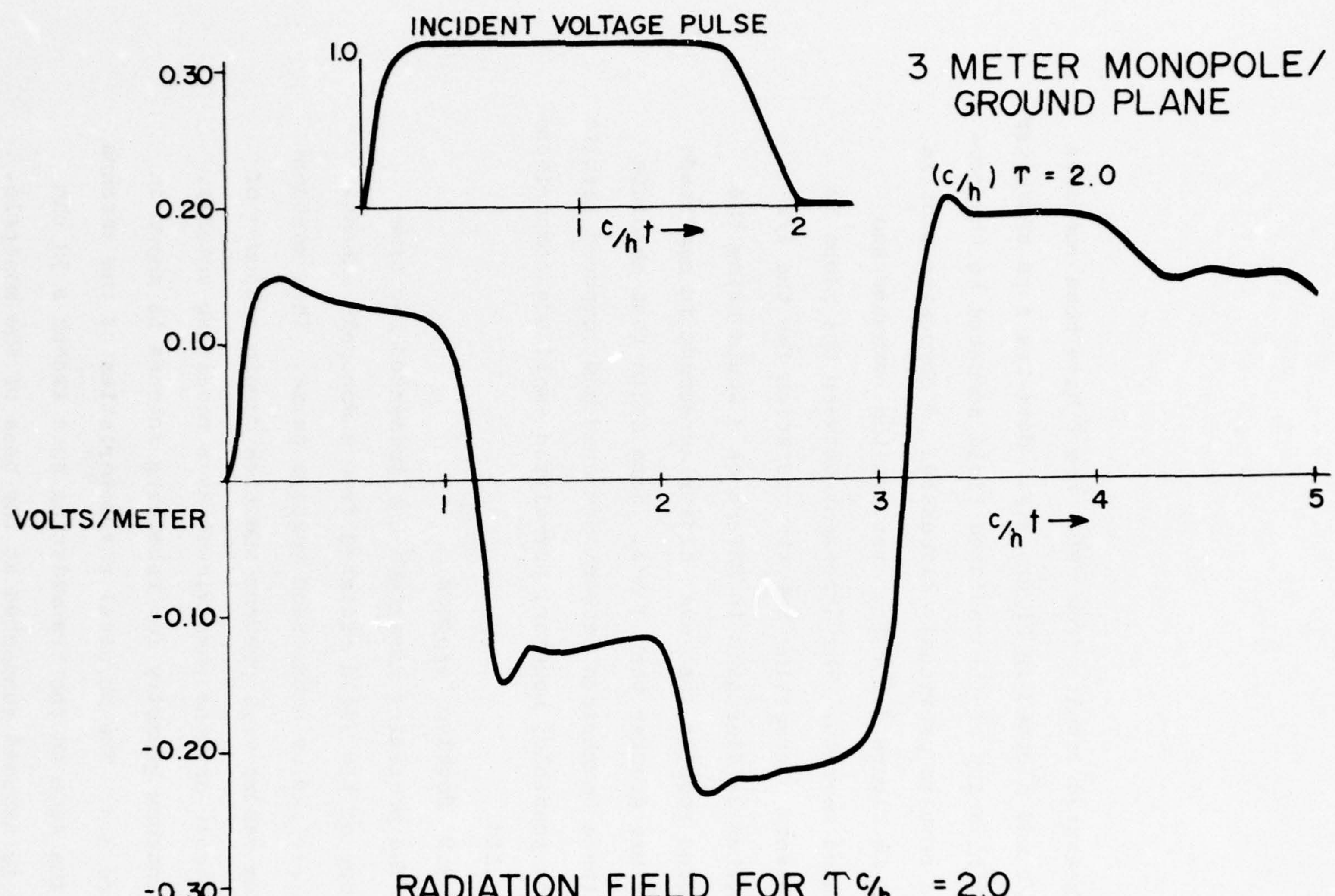
Since both computed and measured results have been plotted in h/c, direct comparison is possible. The two results have been reproduced in overlay format and are shown in Figure 3.5. Direct amplitude comparison between the two traces is not possible; however, amplitude ratios between pulses in each trace will yield relative results. In general, the relative amplitudes compare very well with experimental results. As pointed out by Schmitt, et al., Page 122, corrections are required in the experimental results to account for the 1/R variation due to base and end point displacement. These required corrections will bring the amplitudes into closer agreement. Note the zero crossovers and details of the pulses even after the third pulse. The agreement is excellent and provides a verification for the computed results. The disparity at time t=3 is due to the physical path difference between the base and tip to the field point. In the computation, this difference is neglected since $R \to \infty$ for far field calculations.



COMPUTED VS MEASURED RADIATION FIELD

Figure 3.5

The computed results for the rectangular pulse excitation have been plotted and are shown in Figure 3.6. Note the input pulse which is now twice as long as the monopole time-length. While the effects of isolated points of radiation are not as obvious in this case as the previous one, the radiation field can be relatively simply described. At time $t = 0$, the leading edge of the voltage pulse emerges at the base of the monopole and a step radiation occurs. At time $t = h/c$, the leading edge of the pulse on the antenna is reflected from the tip and a step radiation of opposite polarity and twice the amplitude occurs. Consequently, we note negative polarity radiation from h/c to $2\ h/c$. At this time the trailing edge of the voltage pulse is reflected from the tip while the leading edge is being reflected from the base with smaller amplitude. These two sources of radiation which occur at the same time result in a negative radiation of approximately twice the amplitude of the initial radiation and last approximately h/c seconds, the time for the trailing edge of the excitation pulse and reflection to arrive back at the base of the monopole. Continuing in this fashion and noting the locations and reflections of the leading and trailing edges of the voltage excitation predicts the radiated field from $3\ h/c$ to $5\ h/c$.



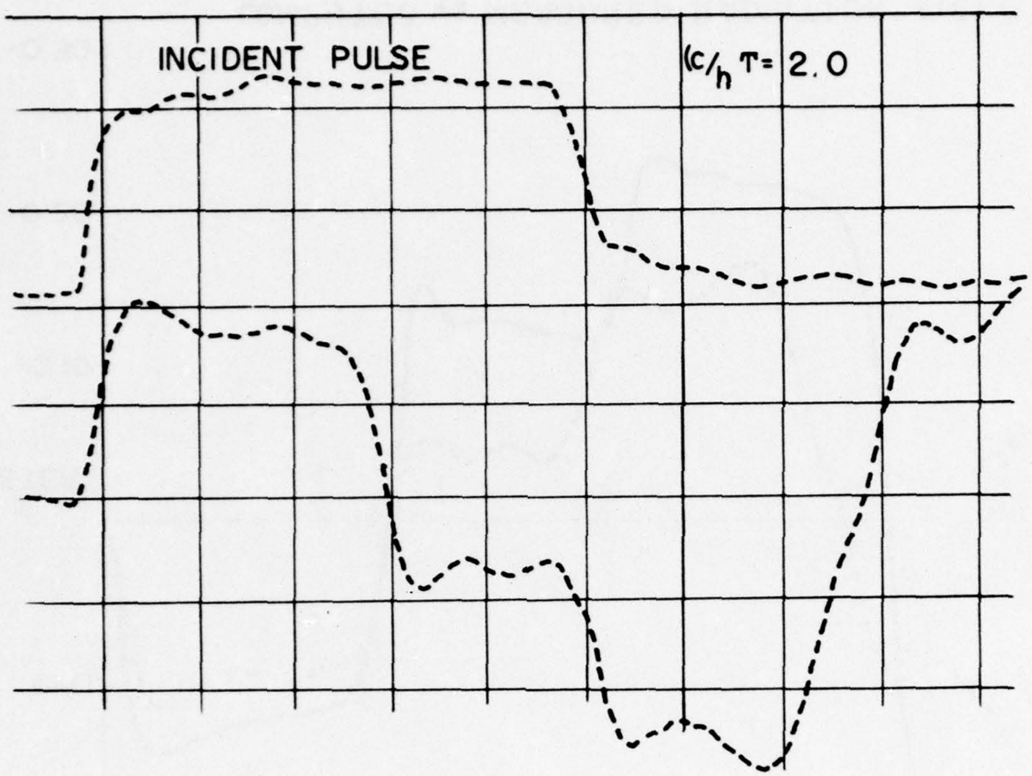
RADIATION FIELD FOR $T c/h = 2.0$

Figure 3.6

The measured results from Reference 2 have been rescaled in h/c and plotted in Figure 3.7. Note the zero crossover and similarity of the radiated field compared to the computed results previously discussed. A composite plot is shown in Figure 3.8 which overlays the computed and measured results. The agreement between the plots is excellent. Correcting the tip radiation for the 1/R variation as discussed in Reference 3 would bring the computed results into even closer agreement in amplitude for times greater than 3 h/c. Note again that absolute amplitude comparison between measured and computed results is not possible; however, normalized amplitude comparisons are valid.

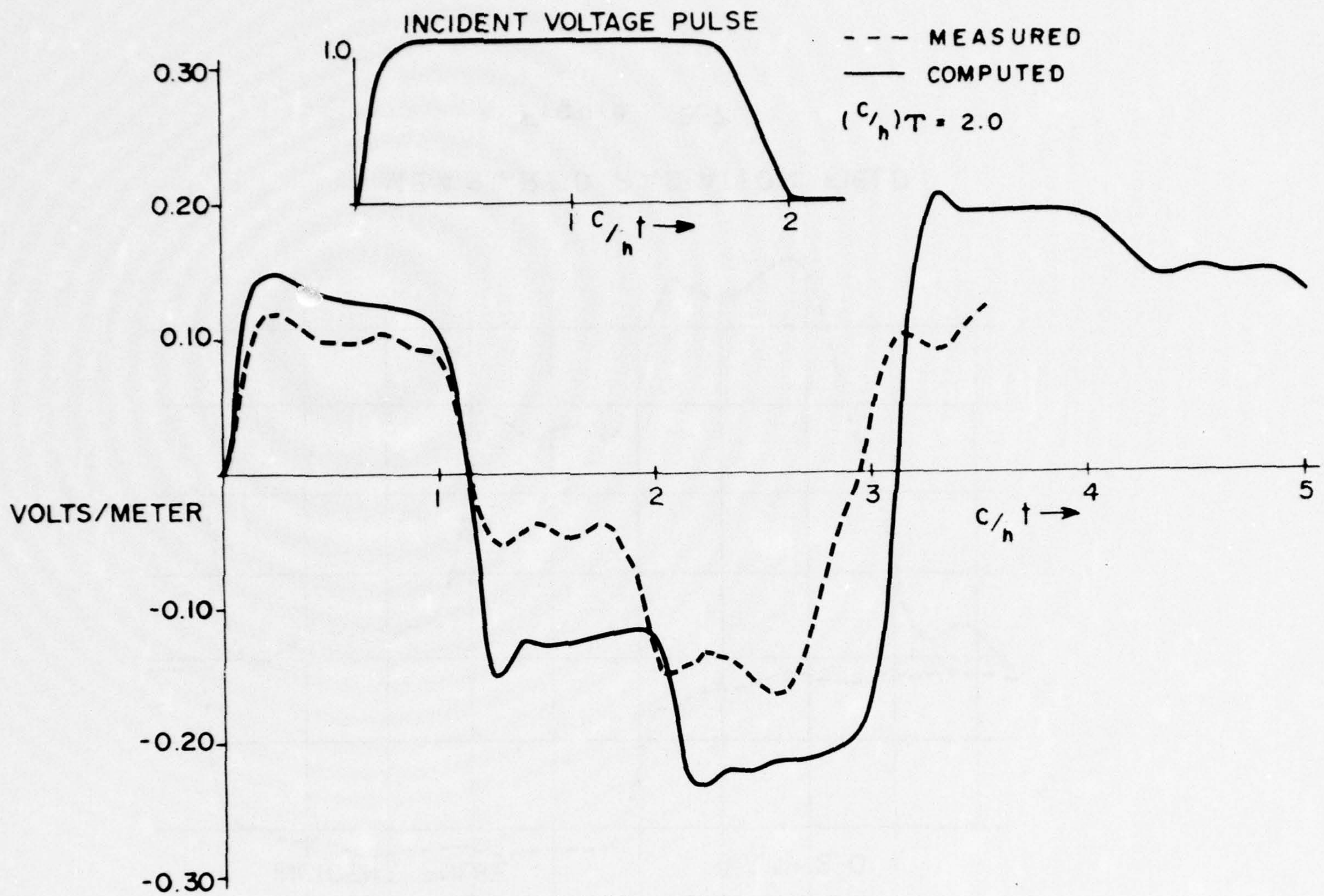
3.2 Receive Response.

The preceding paragraph has presented the time history of the field radiated from a monopole antenna when excited by a baseband voltage pulse. This section of the report will present the time domain response of a monopole antenna when viewed as a receiving antenna. The antenna geometry for receiving antenna is shown in Figure 3.9. The physical characteristics of the antenna are the same as the transmitting case except a 50 ohm load is assumed connected at the base of the monopole.



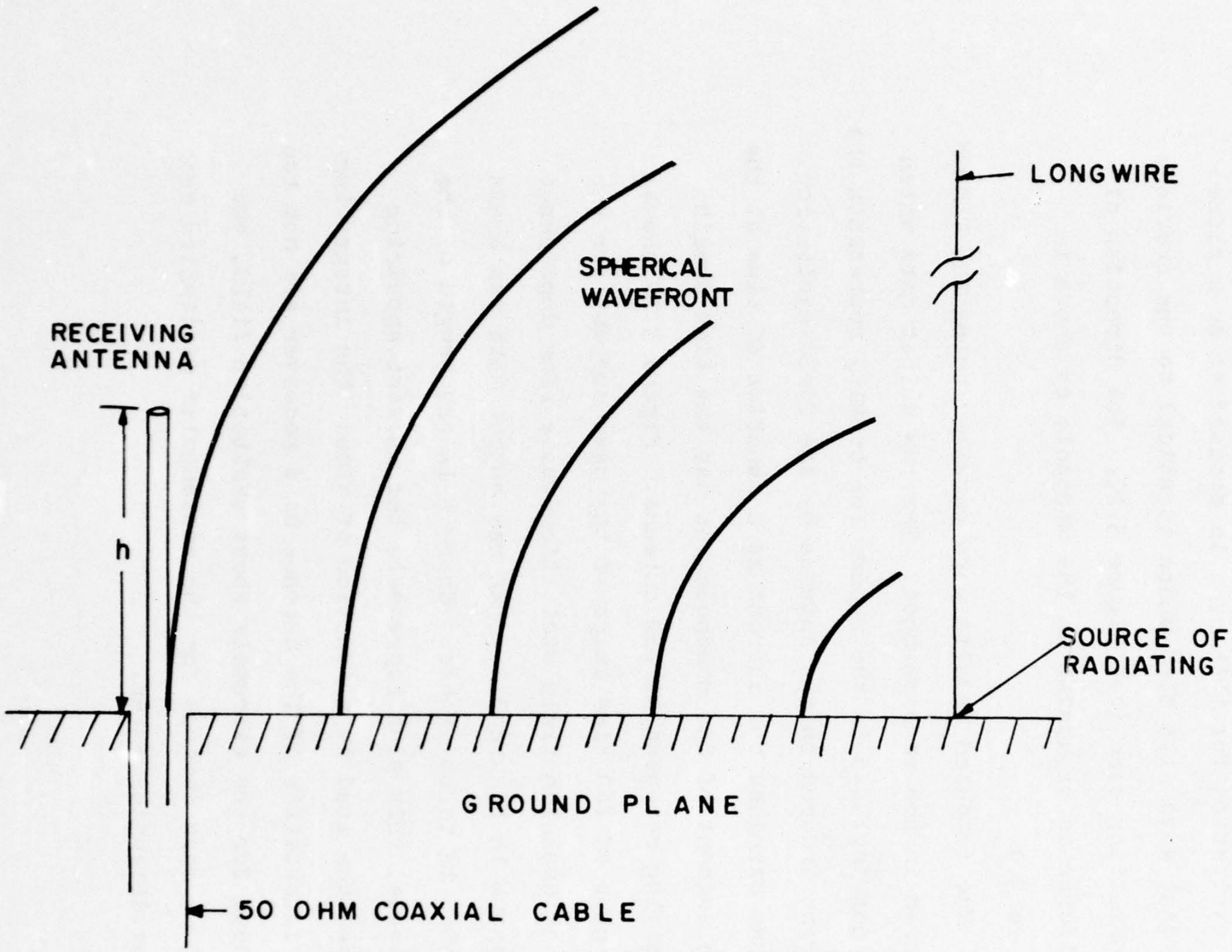
MEASURED RADIATION FIELD

Figure 3.7



COMPUTED vs MEASURED RADIATION FIELD

Figure 3.8

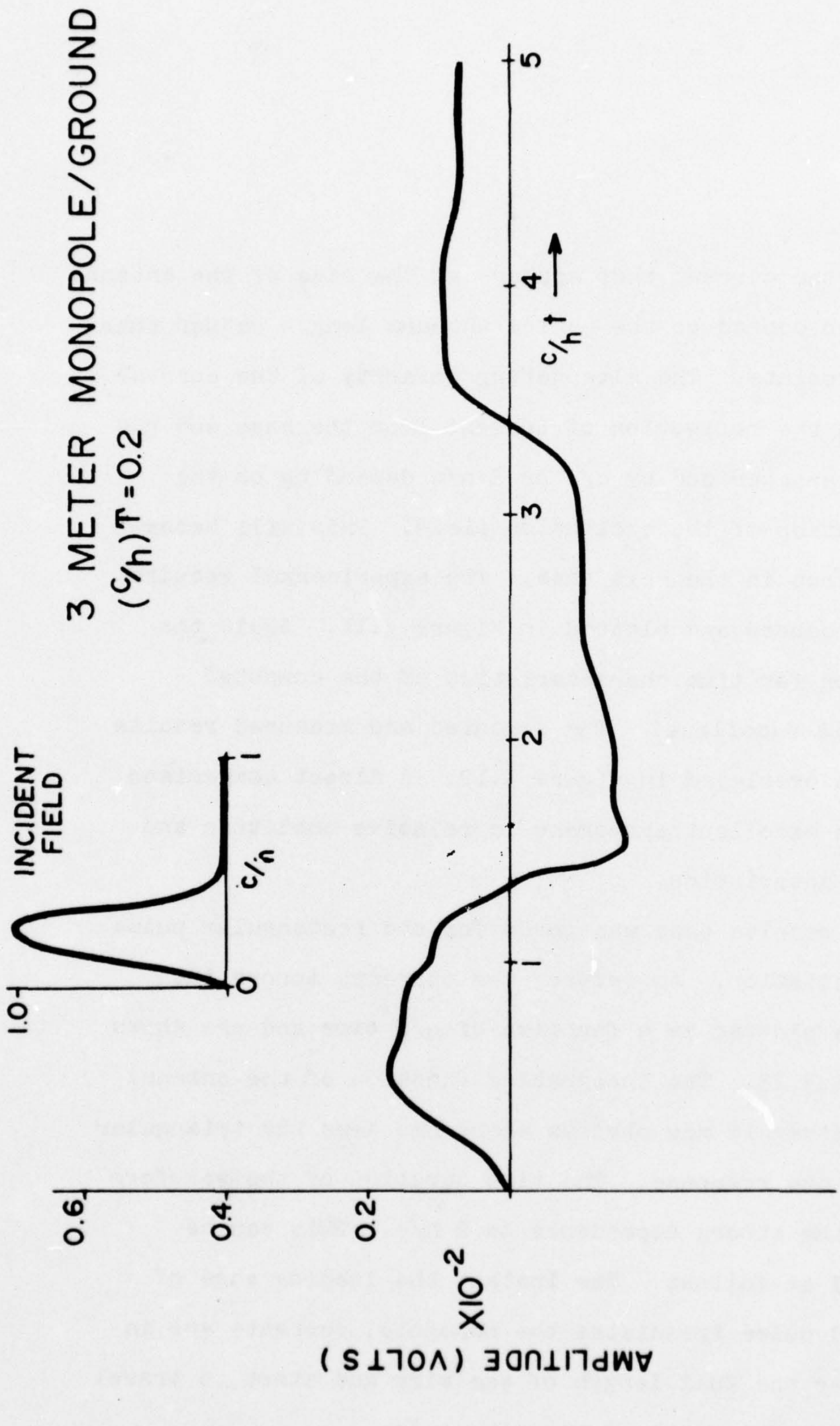


RECEIVING ANTENNA GEOMETRY

Figure 3.9

The excitation for this case is assumed to be a plane incident wave with time shape identical to the previous transmitting case (see Figure 3.2). The direction of incidence is broadside to the monopole as shown in Figure 3.9.

The computed results were obtained using the TIMDOM program in the receive mode. The raw output data which was not included in the report for brevity represents the current induced on the antenna by the field excitation. If one examines the current as a function of time at the base segment of the monopole, he has the time domain receiving response of the antenna. Figure 3.10 shows the plot of 160 time steps of the base current for the narrow Gaussian field excitation whose time dependence is shown in Figure 3.2. The raw output data was again plotted in terms of h/c. Where h is the length of the monopole, this plot represents the current appearing across the load as a function of time. The integrating characteristics of the antenna as a receiver are not too obvious for the extremely short excitation field, but will be more obvious for the rectangular excitation case to be discussed later.

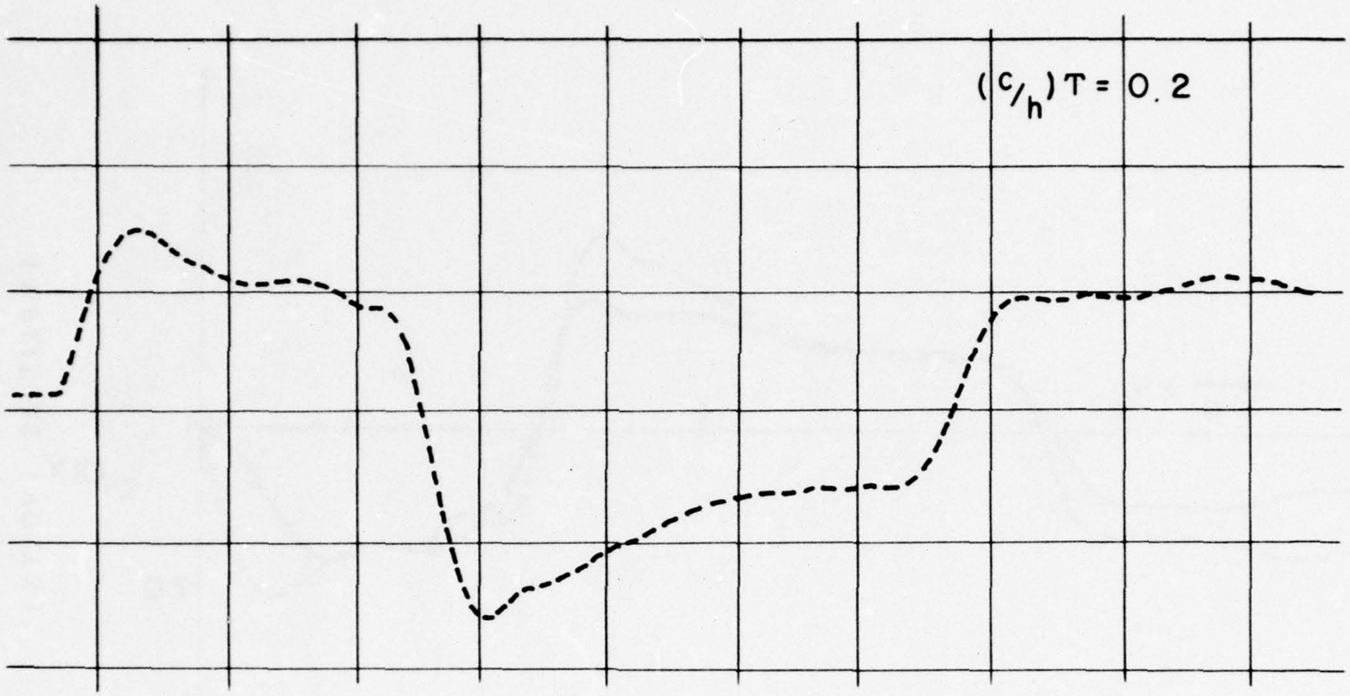


RECEIVE WAVEFORM

Figure 3.10

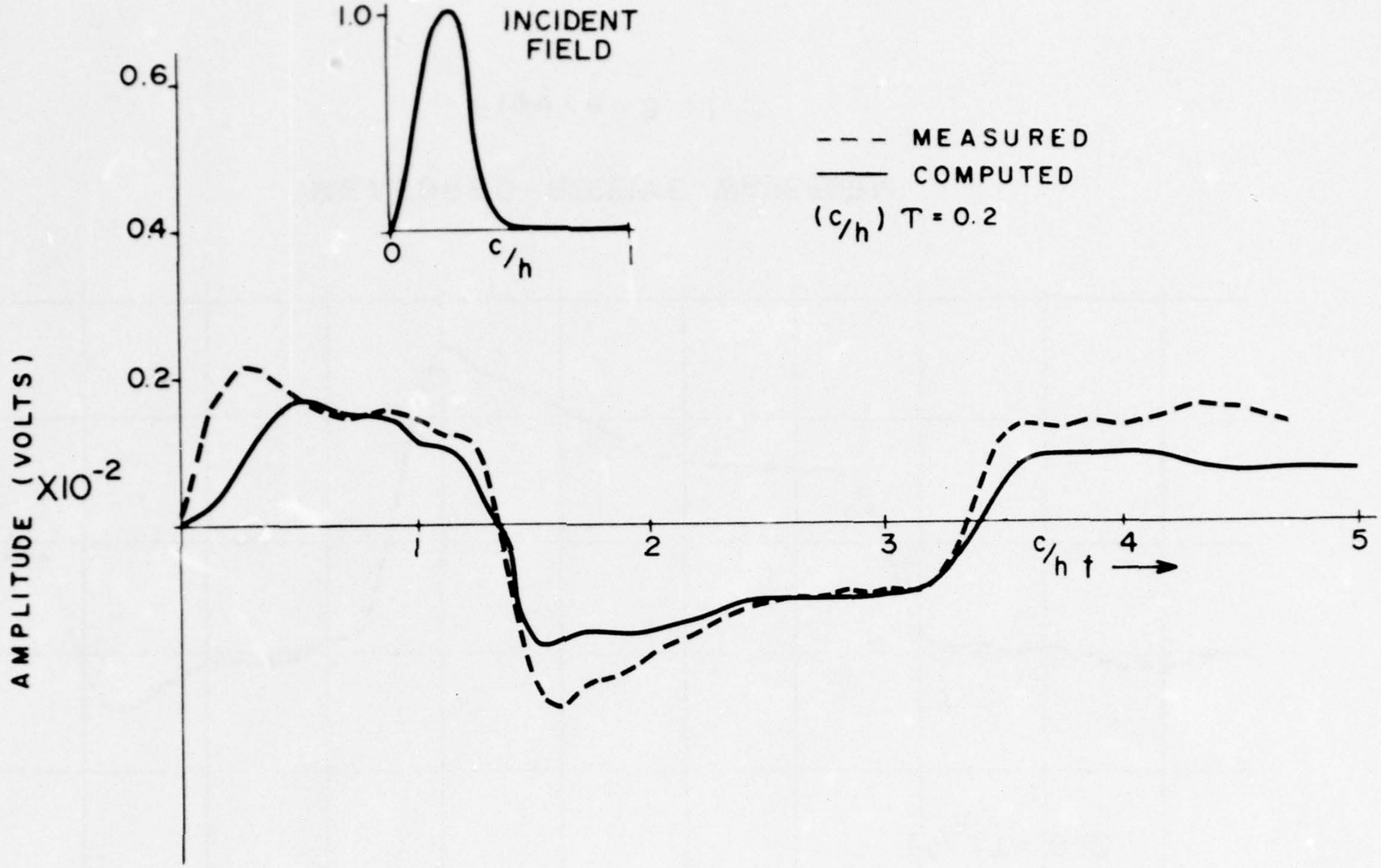
However, the current that appears at the base of the antenna is seen to depend on the entire antenna length rather than isolated points. The alternating polarity of the current is due to the reflection of current from the base and tip and is characterized by h/c or 2 h/c depending on the time duration of the excitation field. This will become more obvious in the next case. The experimental results were reproduced and plotted in Figure 3.11. Again the comparison for time characteristics of the computed results is excellent. The computed and measured results have been overlayed in Figure 3.12. A direct comparison shows the excellent agreement in relative amplitude and time characteristics.

The receive case was rerun for the rectangular pulse field excitation. As before, the currents across the load were plotted as a function of h/c time and are shown in Figure 3.13. The integrating function of the antenna as a receiver is now obvious where one sees the triangular shape of the response. The time duration of the waveform reveals the strong dependence on 2 h/c. This can be explained as follows. The instant the leading edge of the field pulse irradiates the monopole, currents are induced over the full length of the wire and start to travel in both directions along the wire axis.



MEASURED RECEIVE WAVEFORM

Figure 3.11



COMPUTED VS MEASURED RECEIVE WAVEFORM

Figure 3.12

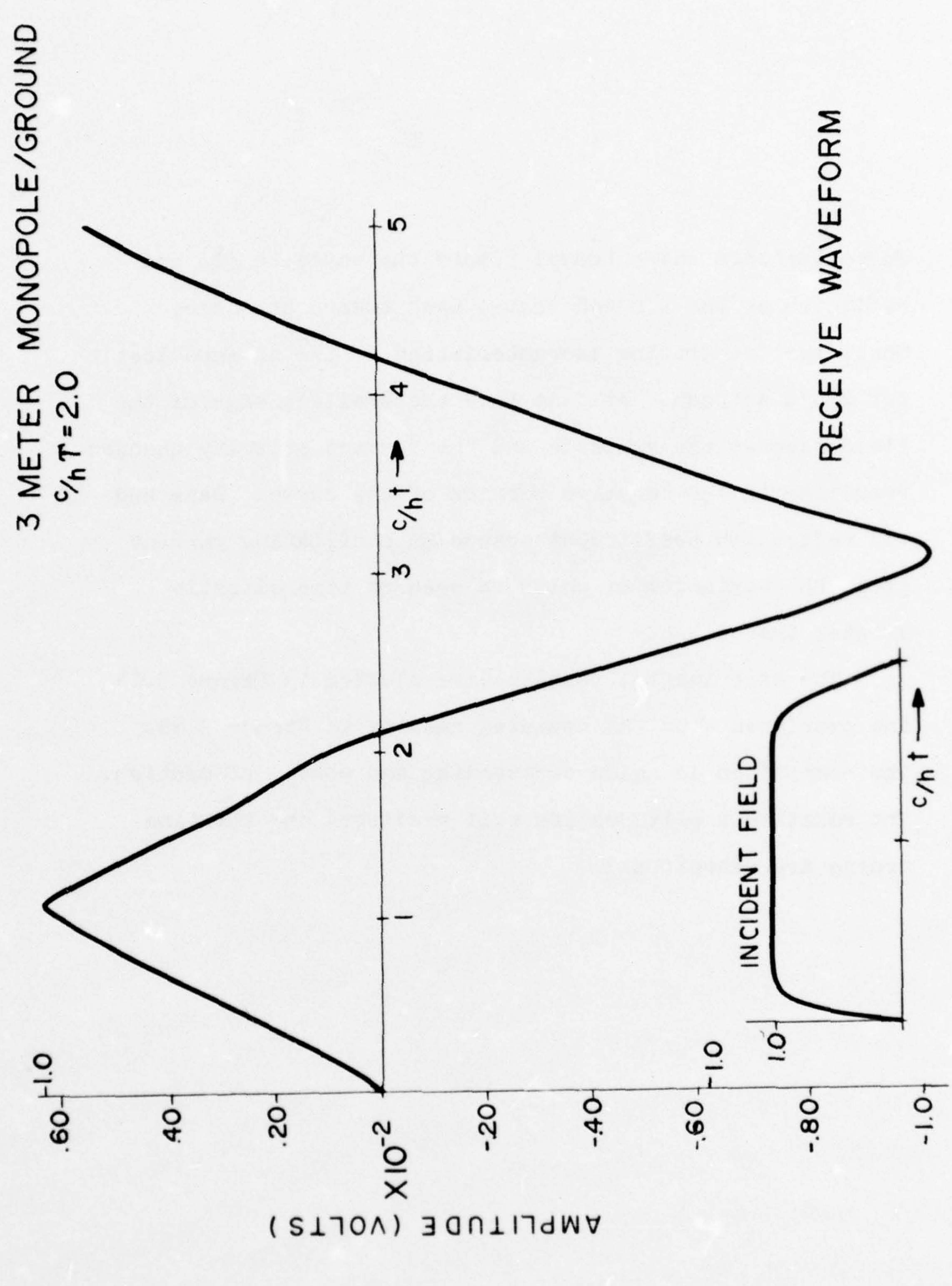
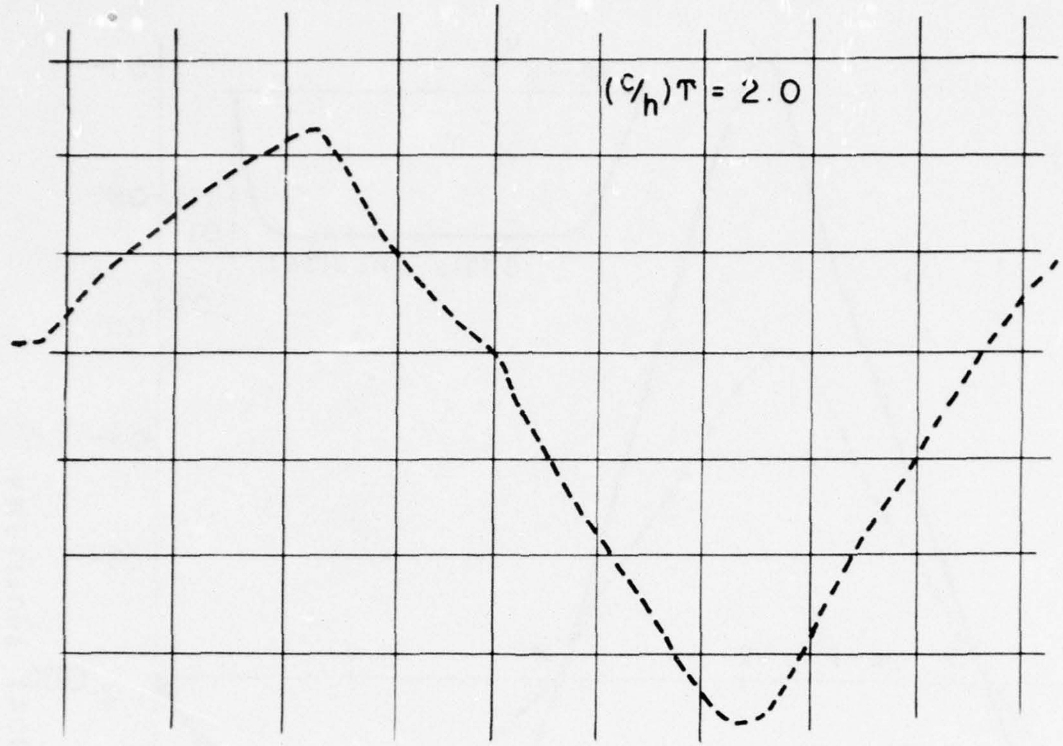

3 METER MONOPOLE/GROUND
c/h T = 2.0
RECEIVE WAVEFORM
INCIDENT FIELD
1.0
c/h t →
AMPLITUDE (VOLTS)
X10^-2
.60
.40
.20
-.20
-.40
-.60
-.80
-1.0
1
2
3
4
5

Figure 3.13

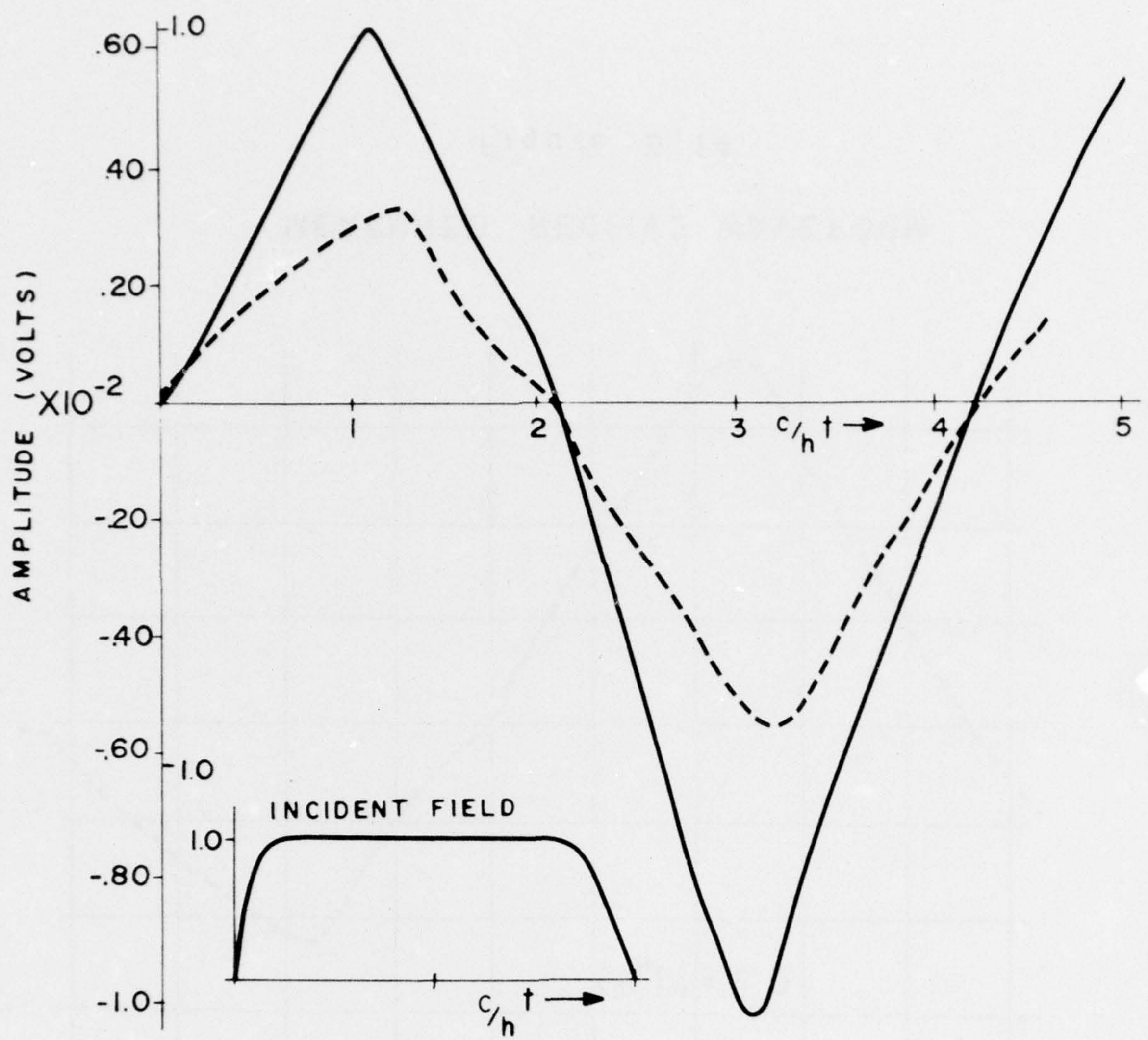
Those currents which travel toward the monopole tip are reflected at the tip and travel back toward the base. Thus, the integrating characteristics of the antenna last for 2 h/c seconds. At that time the trailing edge of the field reaches the monopole and the current polarity changes resulting in the negative portion of the curve. Base and tip reflection coefficients cause an oscillating current plot, the beginning of which is seen at time slightly greater than 4 h/c.

The experimental results were plotted in Figure 3.14 and overlayed with the computed results in Figure 3.15. The comparison is again outstanding and worthy of mention. The relative amplitudes are well predicted and the time traces are exceptional.



MEASURED RECEIVE WAVEFORM

Figure 3.14



COMPUTED VS MEASURED RECEIVE WAVEFORM
Figure 3.15

SECTION IV Conclusions and Recommendations

A computer routine based upon direct time domain calculations has been presented and discussed. A walk-through description of inputting the data and interpreting the output has been provided through an example case to familiarize the reader with the ease of user interface. The computer results have been discussed and critically compared with experimentation. The comparison is exceptional and establishes the validity of the program. The program can be used to predict the transmitting and receiving properties of the class of thin wire antennas directly in the time domain. In addition, although not done in this report, the scattered field can also be obtained through the program. The scattered field is the field reradiated from the antenna when irradiated by a field. It results from the radiation caused by the currents induced on the structure due to the field irradiation.

The utility of direct time domain solutions for pulse excitations is receiving more and more consideration in antenna and scatterer analysis. Although this report has not attempted to establish the utility of direct time domain solutions, its importance as an analytical tool

should not be overlooked. It offers considerably more information concerning the broadband frequency performance of an antenna than does c.w. analysis. In addition, the Fourier transform of the time response (an option in the TIMDOM program) can be used to obtain the frequency response of the antennas modeled. If one is concerned with short pulse excitations and time waveform distortions the time domain offers a direct evalution of antenna response. However, if characteristics such as beam width, impedance, admittance, etc., are important, then the frequency domain offers analytical advantages. Nevertheless, it is apparent that direct time domain solutions offer a new look at antenna performance and should be useful to the antenna analyst and designer. A Fortran listing of the TIMDOM program is included in this report as Appendix A. It must be reemphasized that the computer routine was prepared for user-oriented ease of operation and can be used with only minimal theoretical or numerical techniques understanding.

SECTION V References

(1) "Time Domain Antenna Study" (Final Report), NRL Contract N00014-73-C-0099

(2) Harrington, R.F., "Field Computations By Method of Moments", MacMillan, NY, 1968

(3) Schmitt, H.J., Harrison, C.W., and Williams, Jr. C.S., "Calculated and Experimental Response of Thin Cylindrical Antennas to Pulse Excitation", IEEE Trans. on Antennas & Propagation, Vol. AP-14, March 1966

APPENDICES

The following appendices are included for those interested in implementation of the program on specific machines.

Appendix A contains a Fortran listing of the TIMDOM program which was written by MB Associates under joint sponsorship by the Navy and Air Force under Contracts N00014-73-C-0099 and F30602-72X0008. The listing contains all subroutines in alphabetical order and all are written in Fortran.

Appendix B contains a listing of all machine independent diagnostics which appear in the program to aid the user.

Machine dependent subroutines employed in the program which are specific to the Honeywell/GE635 computer, are identified and listed in Appendix C.

APPENDIX A

```
C     TIME DOMAIN ANTENNA MODELING PROGRAM
      COMMON /DATA/  N,NP,X(200),Y(200),Z(200),SI(200),BI(200),ALP(200),
     1    BET(200),ICON1(200),ICON2(200),ITAG(200),IPX,IPY,IPZ
      COMMON /SCOMP/  SX(200),SY(200),SZ(200)
      COMMON /CONST/  CDT,VEL,DT,NTSTEP
      COMMON /LOAD/  RES(200),ELD(200),CAP(200),ILOAD
      COMMON /IOFLG/  NCFMX,NCQMX,IOC,IOCQ,NTMAX,NBOUT,JP1,JP2
      COMMON /MATPAR/  NBLOKS,NPBLK,NLAST,INX
      COMMON /ARRAY/  CQ(6400)
      COMMON /ESORC/  ESORC(1024),IFST,NTRAN,MTRAN,NTNX,DFRQ,ENIN,ENRD,I
     1FEN
      COMMON /ASORC/  EMAG(10),ISRC(10),NSRC
      COMMON /CMAT/  CURF(1600),IP(200),IX(200)
      DIMENSION COM(13,5), ITST(8)
      INTEGER  HX,HY,HZ
      DATA   HX,HY,HZ/1HX,1HY,1HZ/
      DATA   ITST/2HCM,2HCE,2HAT,2HRF,2HPC,2HNX,2HEN,2HEB/
      RTM1=0.
      RTM2=0.
      RTM3=0.
      CALL SECOND (RTM1)
      NCFMX=17600
      NCQMX=6400
      IRESRV=1600
      VEL=2.998E+8
      PI=3.1415926
      TA=PI/180.
      TD=180./PI
      NTNX=1024
    1 PRINT 57
      CALL SECOND (RTM2)
      IFEN=0
      IFST=0
      JCOM=0
C
C     READ AND PRINT COMMENT CARDS
C
    2 JCOM=JCOM+1
      IF (JCOM.GT.5) JCOM=5
      READ 58,  INA,(COM(I,JCOM),I=1,13)
      PRINT 59,  (COM(I,JCOM),I=1,13)
      IF (INA.EQ.ITST(1)) GO TO 2
      IF (INA.EQ.ITST(2)) GO TO 3
      PRINT 77
      STOP
C
C     READ AND PRINT FIRST DATA CARD
C
    3 READ 63,  DT,TMAX,NTSTEP,IPX,IPY,IPZ,ILOAD,JP1,JP2
      PRINT 64,  DT,TMAX,NTSTEP,IPX,IPY,IPZ,ILOAD,JP1,JP2
      CALL IFEH (IPX)
      CALL IFEH (IPY)
      CALL IFEH (IPZ)
C
C     CALL DATAGN TO SET UP STRUCTURE GEOMETRY
C
```

```
      CALL    DATAGN
      NP=N
      PRINT 60
      PRINT 61
      SSUM=0.
      SMAX=0.
      DO 4 I=1,N
      AD=ALP(I)*TD
      BD=BET(I)*TD
      SII=SI(I)
      PRINT 62,  I,X(I),Y(I),Z(I),SII,AD,BD,BI(I),ICON1(I),I,ICON2(I),
     1 ITAG(I)
      SSUM=SSUM+SII
      IF (SII.GT.SMAX) SMAX=SII
 4    CONTINUE
      PRINT 65,  N,SSUM,SMAX
C
C     DETERMINE TIME STEPPING INTERVAL, DT
C
      IF (DT.GT.1.E-25) GO TO 5
      DT=SMAX/VEL
      SII=10.**(INT(ALOG10(DT))-3)
      DT=AINT(DT/SII+.999)*SII
 5    NTMAX=INT(TMAX/DT)+1
      IF (NTMAX.GT.1) GO TO 6
      IF (NTSTEP.EQ.0) GO TO 39
      GO TO 8
 6    IF (NTSTEP.GT.0) GO TO 7
      NTSTEP=NTMAX
      GO TO 8
 7    IF (NTMAX.LT.NTSTEP) NTSTEP=NTMAX
 8    TMAX=DT*FLOAT(NTSTEP-1)
      NTRAN=NTSTEP
      IF (NTRAN.GT.NTNX) NTRAN=NTNX
      CDT=DT*VEL
      PRINT 73,  DT,TMAX,NTSTEP
      IF (JP1.LT.1) GO TO 9
      IF (JP2.LT.JP1) JP2=JP1
      GO TO 10
 9    JP1=1
      JP2=N
 10   PRINT 66
      IF (IPX) 11,13,12
 11   PRINT 67,  HX
      GO TO 13
 12   PRINT 68,  HX
 13   IF (IPY) 14,16,15
 14   PRINT 67,  HY
      GO TO 16
 15   PRINT 68,  HY
 16   IF (IPZ) 17,19,18
 17   PRINT 67,  HZ
      GO TO 19
 18   PRINT 68,  HZ
 19   IF (ILOAD.NE.0) PRINT 69
      IF (ILOAD.EQ.0) PRINT 70
      PRINT 71,  JP1,JP2
      IF (ILOAD.EQ.0) GO TO 24
C
C     READ IMPEDANCE LOADING CARDS
```

```
C
      PRINT 74
      DO 20 I=1,N
      RES(I)=0.
      ELD(I)=0.
20    CAP(I)=0.
21    READ 75,  J,ITG,II,RESI,ELDI,CAPI
      I=ISEGNO(ITG,II)
      IF (ABS(CAPI).LT.1.E-20) GO TO 22
      PRINT 76,  ITG,II,I,RESI,ELDI,CAPI
      GO TO 23
22    PRINT 76,  ITG,II,I,RESI,ELDI
23    RES(I)=RESI
      ELD(I)=ELDI
      CAP(I)=CAPI
      IF (J.NE.0) GO TO 21
24    CALL   CONSET
C
C     CALL ESET (ENTRY POINT OF EINCF) TO READ EXCITATION DATA CARDS
C
      CALL ESET (RESI)
      CALL FBLOCK (NBLOKS,NPBLK,NLAST,IRESRV,N,INX)
      CALL COFS (CURF,CURF,N)
      CALL FACIO (CURF,CURF,N,IX,IP)
      CALL SECOND (RTM3)
      RTM=RTM3-RTM2
      PRINT 46,  RTM
C
C     SOLVE FOR STRUCTURE CURRENTS
C
      CALL TSOL (CURF,IX,IP,N)
      CALL SECOND (RTM)
      RTM=RTM-RTM3
      PRINT 47,  RTM
25    READ 40,  INA,ITM1,ITM2,ITM3,ITM4,TM1,TM2,TM3,TM4
      CALL SECOND (RTM)
      RTM=RTM-RTM1
      PRINT 48,  RTM
      PRINT 41,  INA,ITM1,ITM2,ITM3,ITM4,TM1,TM2,TM3,TM4
      IF (INA.EQ.ITST(3)) GO TO 26
      IF (INA.EQ.ITST(4)) GO TO 27
      IF (INA.EQ.ITST(5)) GO TO 28
      IF (INA.EQ.ITST(6)) GO TO 1
      IF (INA.EQ.ITST(7)) STOP
      IF (INA.EQ.ITST(8)) GO TO 29
      PRINT 42
      STOP
C
C     COMPUTE ANTENNA INPUT ADMITTANCE
C
26    I=ISEGNO(ITM1,ITM2)
      CALL ANTRAN (I,TM1,ITM3)
      GO TO 25
C
C     COMPUTE RADIATED FIELDS
C
27    CALL RFPAT (ITM1,ITM2,ITM3,ITM4,TM1,TM2,TM3,TM4)
      GO TO 25
C
C     PUNCH CURRENTS
```

```
C
28    I=ISEGNO(ITM1,ITM2)
      CALL CGET (I,CURF,NTRAN)
      PUNCH 43,  (COM(J,1),J=1,13)
      PUNCH 44,  NTSTEP,I,ITM1,ITM2
      PUNCH 45,  (CURF(J),J=1,NTSTEP)
      GO TO 25
C
C     COMPUTE ENERGY BUDGET
C
29    PRINT 49
      ITM2=0
      ENIN=0.
      ENLS=0.
      IF (NSRC.LT.1) GO TO 33
      IF (IFST.NE.0) GO TO 32
      IFEN=1
      PRINT 50
      DO 31 I=1,NSRC
      IS=ISRC(I)
      VLT=EMAG(I)*SI(IS)
      CALL CGET (IS,CURF,NTRAN)
      SUM=0.
      DO 30 J=1,NTRAN
      RESI=ESORC(J)
      IF (RESI.LT.1.E-20) GO TO 30
      SUM=SUM+RESI*CURF(J)
30    CONTINUE
      SUM=SUM*DT*VLT
      PRINT 51,  I,IS,SUM
31    ENIN=ENIN+SUM
      GO TO 33
32    PRINT 56
33    IF (ILOAD.EQ.0) GO TO 37
      DO 36 I=1,N
      IF (RES(I).LT.1.E-20) GO TO 36
      IF (ITM2.EQ.1) GO TO 34
      PRINT 52
      ITM2=1
34    CALL CGET (I,CURF,NTRAN)
      SUM=0.
      DO 35 J=1,NTRAN
35    SUM=SUM+CURF(J)*CURF(J)
      SUM=SUM*DT*RES(I)
      PRINT 53,  I,RES(I),SUM
      ENLS=ENLS+SUM
36    CONTINUE
37    ENRD=ENIN-ENLS
      PRINT 54,  ENIN,ENLS,ENRD
      IF (ENRD.LT.1.E-20) IFEN=0
      IF (ENIN.GT.1.E-20) GO TO 38
      IFEN=0
      GO TO 25
38    TM1=100.*ENRD/ENIN
      PRINT 55,  TM1
      GO TO 25
39    PRINT 72
      STOP
C
C
```

```
40    FORMAT  (A2,I3,3I5,6E10.3)
41    FORMAT  ( //,1X,15H*** DATA CARD**,A2,I3,3I5,6E12.5)
42    FORMAT  (  //,1X,38HINVALID DATA CARD LABEL AFTER SOLUTION)
43    FORMAT  (13A6)
44    FORMAT  (I5,21HCURRENTS FROM SEGMENT,I5,6X,4HTAG=,I5,2X,10HINCREME
     1NT=,I5)
45    FORMAT  (6E12.5)
46    FORMAT  (  1X,37HTIME FOR INITIALIZATION OF CONSTANTS=,F9.3,5H SEC
     1.)
47    FORMAT  (  1X,26HTIME FOR CURRENT SOLUTION=,F9.3,5H SEC.)
48    FORMAT  (  /,1X,13HRUNNING TIME=,F9.3,5H SEC.)
49    FORMAT  (  ///,36X,25H- - - ENERGY BUDGET - - -)
50    FORMAT  (  //,33X,9HSOURCES -,//,35X,15HSOURCE   SEGMENT,3X,12HENER
     1GY INPUT,/,37X,3HNO.,6X,3HNO.,6X,8H(JOULES))
51    FORMAT  (35X,I5,4X,I5,5X,E11.4)
52    FORMAT  (  //,33X,7HLOADS -,/,35X,7HSEGMENT,3X,10HRESISTANCE,2X,11
     1HENERGY LOSS,/,38X,3HNO.,7X,6H(OHMS),5X,8H(JOULES))
53    FORMAT  (36X,I5,2X,2E12.4)
54    FORMAT  (  /,36X,20HTOTAL ENERGY INPUT =,E11.4,7H JOULES,/,36X,22H
     1ENERGY LOST IN LOADS =,E11.4,7H JOULES,/,36X,23HTOTAL ENERGY RADIA
     2TED =,E11.4,7H JOULES)
55    FORMAT  (  36X,24HTIME DOMAIN EFFICIENCY =,F7.2,8H PERCENT)
56    FORMAT  (  /,59HINPUT ENERGY NOT COMPUTED SINCE SOURCE HAS BEEN TR
     1ANSFORMED)
57    FORMAT  (1H1,   /,28X,42H******************************************
     1,/ /,31X,36HTIME DOMAIN ANTENNA MODELING PROGRAM,//,28X,42H*******
     2***********************************,////)
58    FORMAT  (A2,13A6)
59    FORMAT  (28X,13A6)
60    FORMAT  (////    33X,33H- - - - SEGMENTATION DATA - - - -, //,40X,
     1  21HCOORDINATES IN METERS,//,25X,
     2  50HI+ AND I- INDICATE THE SEGMENTS BEFORE AND AFTER I,//)
61    FORMAT  (2X,4HSEG.,3X,26HCOORDINATES OF SEG. CENTER,5X,4HSEG.,
     1  5X,18HORIENTATION ANGLES,4X,4HWIRE,5X,15HCONNECTION DATA,4X,
     2     3HTAG,/,2X,3HNO.,7X,1HX,9X,1HY,9X,1HZ,7X,6HLENGTH,5X,5HALPHA,
     3  5X,4HBETA,6X,6HRADIUS,4X,2HI-,4X,1HI,5X,2HI+,4X,3HNO.)
62    FORMAT  (1X,I5,4F10.5,1X,3F10.5,3I6,2X,I5)
63    FORMAT  (2E10.3,I5,2X,3A1,3I5)
64    FORMAT  (  //,1X,21H*** FIRST DATA CARD**,2E12.5,I5,2X,3A1,3I5)
65    FORMAT  (  //,1X,19HNUMBER OF SEGMENTS=,I7,/,1X,18HTOTAL WIRE LENG
     1TH=,F10.5,3H M.,/,1X,23HMAXIMUM SEGMENT LENGTH=,F10.5,3H M.)
66    FORMAT  (  ///,1X,21HOPTIONS SELECTED- - -,/)
67    FORMAT  (  5H THE ,A1,37H=0 PLANE IS A MAGNETIC SYMMETRY PLANE)
68    FORMAT  (  5H THE ,A1,38H=0 PLANE IS AN ELECTRIC SYMMETRY PLANE)
69    FORMAT  (  1X,27HMODEL HAS IMPEDANCE LOADING)
70    FORMAT  (  1X,19HMODEL IS NOT LOADED)
71    FORMAT  (  1X,37HCURRENTS WILL BE PRINTED FOR SEGMENTS,I5,8H THROU
     1GH ,I5)
72    FORMAT  (1X,25HZERO TIME STEPS REQUESTED)
73    FORMAT  (  ///,1X,40H****************************************,//,1
     1X, 23HTIME STEPPING INTERVAL=,E12.5,5H SEC.,/,1X,13HMAXIMUM TIME=,
     2   E12.5,5H SEC.,/,1X,21HNUMBER OF TIME STEPS=,I6,//,41H *********
     3********************************)
74    FORMAT  (  ///,35X,33H- - - - STRUCTURE LOADING - - - -,//,23X,
     1   60HTAG  INCREMENT  SEG.   RESISTANCE   INDUCTANCE   CAPACITANCE
     2,/ ,23X,3HNO.,14X,3HNO.,5X,6H(OHMS),6X,8H(HENRYS),6X,8H(FARADS))
75    FORMAT  (3I5,5X,3E10.3)
76    FORMAT  (21X,I5,5X,I5,2X,I5,3X,E10.4,3X,E10.4,4X,E10.4)
77    FORMAT  (1X,34HINCORRECT LABEL FOR A COMMENT CARD)
      END
```

```
      SUBROUTINE ANTRAN (ISEG,ANOR,IPCH)
C
C     ANTRAN COMPUTES AND PRINTS THE INPUT ADMITTANCE AND IMPEDANCE AT
C     SEGMENT ISEG
C
      COMMON /DATA/  N,NP,X(200),Y(200),Z(200),SI(200),BI(200),ALP(200),
     1    BET(200),ICON1(200),ICON2(200),ITAG(200),IPX,IPY,IPZ
      COMMON /ESORC/  ESORC(1024),IFST,NTRAN,MTRAN,NTNX,DFRQ,ENIN,ENRD,I
     1FEN
      COMMON /CMAT/  CURF(1600),IP(200),IX(200)
      COMMON /CONST/  CDT,VEL,DT,NTSTEP
      COMMON /ASORC/  EMAG(10),ISRC(10),NSRC
      COMMON /SCRATM/ PWR(512)
      COMPLEX    YMIT,ZPED,ZNOR
      DATA   TSMIN/5./
      IF (IFST.EQ.1) GO TO 2
C
C     FOURIER TRANSFORM SOURCE
C
      MTRAN=ALOG(FLOAT(NTRAN))/.69314718+1.5
      CALL ITOF (ESORC,NTRAN,MTRAN,1,1,NTNX,NTNR)
      IFST=1
      DFRQ=1./(DT*FLOAT(2**MTRAN))
      NTNR=NTNR/2
      DO 1 I=1,NTNR
1     PWR(I)=0.
2     CALL CGET (ISEG,CURF,NTRAN)
C
C     FOURIER TRANSFORM CURRENT
C
      CALL ITOF (CURF,NTRAN,MTRAN,0,0,NTNX,NTNR)
      INOR=0
      IF (ABS(ANOR).GT.1.E-20) INOR=1
      IF (NSRC.LT.1) GO TO 4
      DO 3 I=1,NSRC
      I1=I
      IF (ISRC(I).EQ.ISEG) GO TO 5
3     CONTINUE
4     VLT=1.
      ISPR=0
      GO TO 6
5     VLT=EMAG(I1)*SI(ISEG)
      ISPR=1
6     PRINT 10, ISEG,VLT
      IF (INOR.EQ.1) PRINT 11,  ANOR
      PRINT 12
      DF=DFRQ*1.E-6
      F=-DF
      FMAX=1.E-6/(TSMIN*DT)
      NTNR=NTNR/2
      IF (IPCH.NE.0) PUNCH 15
      DO 8 I=1,NTNR
      F=F+DF
      IF (F.GT.FMAX) GO TO 9
      I2=2*I
      I1=I2-1
```

```
      YMIT=CMPLX(ESORC(I1),ESORC(I2))*VLT
      SPCT=CABS(YMIT)
      YMIT=CMPLX(CURF(I1),CURF(I2))/YMIT
      ZPED=1./YMIT
      IF (ISPR.EQ.1) PWR(I)=PWR(I)+.5*VLT*VLT*REAL(YMIT)
      IF (IPCH.NE.0) PUNCH 14, I,F,ZPED,YMIT
      IF (INOR.EQ.0) GO TO 7
      ZNOR=ZPED/ANOR
      PRINT 13,  I,F,ZPED,YMIT,ZNOR,SPCT
      GO TO 8
7     PRINT 14,  I,F,ZPED,YMIT,SPCT
8     CONTINUE
9     RETURN
C
C
10    FORMAT  (  //,24X,50H- - - ANTENNA INPUT IMPEDANCE AND ADMITTANCE
     1- - -,//,31X,23HSOURCE SEGMENT NUMBER =,I5,/,31X,21HPEAK SOURCE VO
     2LTAGE =,E11.4,6H VOLTS)
11    FORMAT  (31X,32HIMPEDANCE NORMALIZATION FACTOR =,E11.4,5H OHMS)
12    FORMAT  (  /,1X,4HSTEP,4X,9HFREQUENCY,6X,16HIMPEDANCE (OHMS),10X,
     1   17HADMITTANCE (MHOS),7X,20HNORMALIZED IMPEDANCE,6X,6HSOURCE,/,
     2 2X,3HNO.,6X,5H(MHZ),3(8X,4HREAL,8X,5HIMAG.,1X),[illegible],8HSPECTRUM)
13    FORMAT  (1X,I4,2E13.4,E12.4,2(E14.4,E12.4),2X,E11.4)
14    FORMAT (1X,I4,2E13.4,E12.4,E14.4,E12.4,28X,E11.4)
15    FORMAT (9X,9HFREQUENCY,9X,9HIMPEDANCE,18X,10HADMITTANCE)
      END
```

```
      FUNCTION ATGN2 (X,Y)
C
C     ATGN2 IS ARCTANGENT FUNCTION MODIFIED TO RETURN 0. WHEN X=Y=0.
C
      IF (X) 3,1,3
1     IF (Y) 3,2,3
2     ATGN2=0.
      RETURN
3     ATGN2=ATAN2(X,Y)
      RETURN
      END
```

```
      SUBROUTINE CGET (ISEG,CSEG,NLM)
C
C     CGET FILLS ARRAY CSEG WITH THE CURRENT ON SEGMENT ISEG FOR ALL
C     TIME STEPS
C
      COMMON /DATA/ N,NP,X(200),Y(200),Z(200),SI(200),BI(200),ALP(200),
     1  BET(200),ICON1(200),ICON2(200),ITAG(200),IPX,IPY,IPZ
      COMMON /ARRAY/ CQ(6400)
      COMMON /IOFLG/ NCFMX,NCQMX,IOC,IOCQ,NTMAX,NBOUT,JP1,JP2
      DIMENSION CSEG(1024)
      II=N*2
      IB=2*ISEG-1
      I=IB-II
      NOCQ=999999
      IF (IOCQ.EQ.0) GO TO 1
      REWIND  11
      READ (11) NOCQ,(CQ(K),K=1,NOCQ)
1     DO 3 J=1,NLM
      I=I+II
      IF (I.LE.NOCQ) GO TO 2
      READ (11) NOCQ,(CQ(K),K=1,NOCQ)
      I=IB
2     CSEG(J)=CQ(I)
3     CONTINUE
      IF (IOCQ.NE.0) REWIND  11
      RETURN
      END
```

```
      SUBROUTINE COFS (CURF,CURL,NDIM)
C
C     COFS SETS UP THE ARRAYS OF INTERACTION COEFFICIENTS FOR BOTH
C     PRESENT TIME AND RETARDED TIME INTERACTION
C
      COMMON /DATA/ N,NP,X(200),Y(200),Z(200),SI(200),BI(200),ALP(200),
     1   BET(200),ICON1(200),ICON2(200),ITAG(200),IPX,IPY,IPZ
      COMMON /SCOMP/ SX(200),SY(200),SZ(200)
      COMMON /CONST/ CDT,VEL,DT,NTSTEP
      COMMON /ARRAY/ CQ(6400)
      COMMON /ARRAX/ QI1,QI2,QI3,EC(200,5),EQ(200,5),IRET(200),BFR(15400
     1)
      COMMON /EMATS/ EMX(3,3),EMY(3,3),EMZ(3,3),QMX(3,3),QMY(3,3),QMZ(3,
     13)
      COMMON /LOAD/ RES(200),ELD(200),CAP(200),ILOAD
      COMMON /IOFLG/ NCFMX,NCQMX,IOC,IOCQ,NTMAX,NBOUT,JP1,JP2
      COMMON /MATPAR/ NBLOKS,NPBLK,NLAST,INT
      DIMENSION CURF(NDIM,NDIM), CURL(1)
      DIMENSION COF(17600), ICOF(17600), SRT(6400), ISRT(6400)
      DIMENSION ALFA(3,3), BETA(3,3)
      EQUIVALENCE (COF,EC), (ICOF,EC), (SRT,CQ), (ISRT,CQ)
      LPX=1
      LPY=1
      LPZ=1
      IF (IPX.NE.0) LPX=2
      IF (IPY.NE.0) LPY=2
      IF (IPZ.NE.0) LPZ=2
      ICNTC=0
      ICNTS=0
      ICTOT=0
      IOC=0
      IOS=0
      ICOF(NCFMX)=9898989
      ISRT(NCQMX)=9898989
      IST=2*N*LPX*LPY*LPZ
      ICMAX=NCFMX-IST
      ISMAX=NCQMX-IST
      IST=1
      NTMAX=0
      IF (INT.EQ.0) GO TO 1
      REWIND 13
      NOBLKS=NBLOKS
      I2=NPBLK*N
      IT=NPBLK
      GO TO 2
1     NOBLKS=1
      IT=N
2     I=0
C
C     BEGIN LOOP OVER FIELD EVALUATION POINTS
C
      DO 30 IOBLKS=1,NOBLKS
      IF (INT.EQ.0) GO TO 3
      IF (IOBLKS.EQ.NOBLKS) IT=NLAST
3     DO 29 IB=1,IT
      I=I+1
```

```
      XI=X(I)
      YI=Y(I)
      ZI=Z(I)
      SXI=SX(I)
      SYI=SY(I)
      SZI=SZ(I)
      DO 4 J=1,N
4     CURF(J,IB)=0.
      IFL=0
C
C     BEGIN LOOP OVER SOURCE SEGMENTS INCLUDING IMAGES
C
      DO 29 JX=1,LPX
      RFX=FLOAT(3-JX*2)
      SFX=RFX
      IF (IPX.LT.0) SFX=1.
      DO 29 JY=1,LPY
      RFY=FLOAT(3-JY*2)
      SFY=RFY
      IF (IPY.LT.0) SFY=1.
      DO 29 JZ=1,LPZ
      RFZ=FLOAT(3-JZ*2)
      SFZ=RFZ
      IF (IPZ.LT.0) SFZ=1.
      RFL=SFX*SFY*SFZ
      NTL=0
      DO 5 J=1,N
      RX=XI-X(J)*RFX
      RY=YI-Y(J)*RFY
      RZ=ZI-Z(J)*RFZ
      R=SQRT(RX*RX+RY*RY+RZ*RZ)
      K=R/CDT+.5
      IF (K.EQ.0) K=1
      IRET(J)=K
      IF (K.GT.NTL) NTL=K
      DO 5 M=1,5
      EC(J,M)=0.
5     EQ(J,M)=0.
      DO 21 J=1,N
      RX=XI-X(J)*RFX
      RY=YI-Y(J)*RFY
      RZ=ZI-Z(J)*RFZ
      R2=RX*RX+RY*RY+RZ*RZ
      R=SQRT(R2)
      K=IRET(J)
      TAU=DT*FLOAT(K)
      SXJ=SX(J)*RFX
      SYJ=SY(J)*RFY
      SZJ=SZ(J)*RFZ
      CALL EMAT (RX,RY,RZ,R2,TAU,SXJ,SYJ,SZJ,J)
      DO 6 L=1,3
      DO 6 M=1,3
      ALFA(L,M)=(EMX(L,M)*SXI+EMY(L,M)*SYI+EMZ(L,M)*SZI)*RFL
6     BETA(L,M)=(QMX(L,M)*SXI+QMY(L,M)*SYI+QMZ(L,M)*SZI)*RFL
      JC1=ICON1(J)
      IF (JC1.NE.0) GO TO 7
      KKM=0
      GO TO 11
7     IF (JC1.LT.19000) GO TO 8
      SIG1=FLOAT(JC1-20000)
```

```
      JC1=J
      GO TO 10
8     IF (ICON2(JC1).NE.J) GO TO 9
      SIG1=1.
      GO TO 10
9     IF (ICON1(JC1).NE.J) GO TO 44
      SIG1=-1.
10    KM=IRET(JC1)
      KKM=KM-K
      IF (IABS(KKM).LE.1) GO TO 11
      PRINT 45, I,J,JC1,KKM
      STOP
11    JC2=ICON2(J)
      IF (JC2.NE.0) GO TO 12
      KKP=0
      GO TO 16
12    IF (JC2.LT.19000) GO TO 13
      SIG2=FLOAT(JC2-20000)
      JC2=J
      GO TO 15
13    IF (ICON1(JC2).NE.J) GO TO 14
      SIG2=1.
      GO TO 15
14    IF (ICON2(JC2).NE.J) GO TO 44
      SIG2=-1.
15    KP=IRET(JC2)
      KKP=KP-K
      IF (IABS(KKP).LE.1) GO TO 16
      PRINT 45, I,J,JC2,KKP
      STOP
16    DO 20 M=1,3
      KXX=M+1
      KMX=KKM+KXX
      KPX=KKP+KXX
      KK=K-M+2
      IF (KK.EQ.0) GO TO 18
      IF (JC1.EQ.0) GO TO 17
      EC(JC1,KMX)=EC(JC1,KMX)+ALFA(1,M)*SIG1
      EQ(JC1,KMX)=EQ(JC1,KMX)+BETA(1,M)*SIG1
17    EC(J,KXX)=EC(J,KXX)+ALFA(2,M)
      EQ(J,KXX)=EQ(J,KXX)+BETA(2,M)
      IF (JC2.EQ.0) GO TO 20
      EC(JC2,KPX)=EC(JC2,KPX)+ALFA(3,M)*SIG2
      EQ(JC2,KPX)=EQ(JC2,KPX)+BETA(3,M)*SIG2
      GO TO 20
18    IF (JC1.EQ.0) GO TO 19
      CURF(JC1,IB)=CURF(JC1,IB)-(ALFA(1,3)+BETA(1,3)*QI3)*SIG1
      EQ(JC1,KMX)=EQ(JC1,KMX)+BETA(1,M)*SIG1
19    CURF(J,IB)=CURF(J,IB)-ALFA(2,3)-BETA(2,3)*QI3
      EQ(J,KXX)=EQ(J,KXX)+BETA(2,M)
      IF (JC2.EQ.0) GO TO 20
      CURF(JC2,IB)=CURF(JC2,IB)-(ALFA(3,3)+BETA(3,3)*QI3)*SIG2
      EQ(JC2,KPX)=EQ(JC2,KPX)+BETA(3,M)*SIG2
20    CONTINUE
21    CONTINUE
C
C     ADD TERMS DUE TO LOADING
C
      IF (ILOAD.EQ.0) GO TO 22
      IF (IFL.EQ.1) GO TO 22
```

```
      EC(I,2)=EC(I,2)-ELD(I)*.5/(DT*SI(I))
      EC(I,3)=EC(I,3)+ELD(I)*2./(DT*SI(I))
      CURF(I,IB)=CURF(I,IB)+(RES(I)+ELD(I)*1.5/DT)/SI(I)
      IF (CAP(I).LT.1.E-20) GO TO 22
      CURF(I,IB)=CURF(I,IB)+QI3/(CAP(I)*SI(I))
      EQ(I,4)=EQ(I,4)-1./(CAP(I)*SI(I))
22    NTL=NTL+2
      IF (NTL.GT.NTMAX) NTMAX=NTL
C
C     SORT COEFFICIENTS FOR FIELD EVALUATION SEGMENT I
C
      DO 28 LL=1,NTL
      L=NTL-LL
      IHDG=0
      J=0
23    J=J+1
      IF (J.GT.N) GO TO 28
      K=IRET(J)
      KDIF=K-L
      IF (IABS(KDIF).GT.2) GO TO 23
      IF (IHDG.EQ.1) GO TO 24
      IHDG=1
      ISRT(IST)=-(L*10000+I)
      IST=IST+1
24    KK=J
      KKP=IST
      IST=IST+1
25    SRT(IST)=EC(J,KDIF+3)
      SRT(IST+1)=EQ(J,KDIF+3)
      IST=IST+2
      J=J+1
      IF (J.GT.N) GO TO 26
      K=IRET(J)
      KDIF=K-L
      IF (IABS(KDIF).LT.3) GO TO 25
26    ISRT(KKP)=20000*(J-KK)+2*KK-1
      IF (IST.LT.ISMAX) GO TO 23
      IF (IOS.EQ.1) GO TO 27
      IOS=1
      REWIND 11
27    IST=IST-1
      WRITE (11) IST,(SRT(K),K=1,IST)
      ICNTS=ICNTS+1
      IST=1
      GO TO 23
28    CONTINUE
      IFL=1
29    CONTINUE
      IF (INT.EQ.0) GO TO 30
      WRITE (13) (CURL(I),I=1,I2)
30    CONTINUE
      IF (INT.NE.0) REWIND 13
C
C     SORT ALL COEFFICIENTS ACCORDING TO RETARDED TIME
C
      ISRT(IST)=0
      IF (IOS.EQ.0) GO TO 31
      WRITE (11) IST,(SRT(J),J=1,IST)
      ICNTS=ICNTS+1
      REWIND 11
```

```
31    ICF=1
      DO 40 LL=1,NTMAX
      L=NTMAX-LL
      IST=1
      ISL=9999999
      IF (IOS.EQ.0) GO TO 34
      ISL=0
      REWIND 11
      GO TO 34
32    IF (-INDX/10000.EQ.L) GO TO 33
      IHDG=0
      GO TO 34
33    IHDG=1
      ICOF(ICF)=INDX
      ICF=ICF+1
34    IF (IST.LE.ISL) GO TO 35
      READ (11) ISL,(SRT(J),J=1,ISL)
      IST=1
35    INDX=ISRT(IST)
      IST=IST+1
      IF (INDX) 32,40,36
36    KK=INDX/10000
      IF (IHDG.EQ.0) GO TO 39
      ICOF(ICF)=INDX
      ICF=ICF+1
      KKP=IST
      DO 37 J=1,KK
      COF(ICF)=SRT(KKP)
      ICF=ICF+1
37    KKP=KKP+1
      IF (ICF.LT.ICMAX) GO TO 39
      IF (IOC.EQ.1) GO TO 38
      IOC=1
      REWIND 12
38    ICF=ICF-1
      WRITE (12) ICF,L,(COF(J),J=1,ICF)
      ICTOT=ICTOT+ICF
      ICNTC=ICNTC+1
      ICF=1
39    IST=IST+KK
      GO TO 34
40    CONTINUE
      ICOF(ICF)=0
      IF (IOC.EQ.0) GO TO 41
      WRITE (12) ICF,L,(COF(J),J=1,ICF)
      ICTOT=ICTOT+ICF
      ICNTC=ICNTC+1
      REWIND 12
41    IF (IOS.EQ.1) REWIND 11
      IF (ISRT(NCGMX).NE.9898989) GO TO 42
      IF (ICOF(NCFMX).NE.9898989) GO TO 43
      IF (ICNTC.EQ.0) ICTOT=ICF
      PRINT 49, ICTOT,ICNTS,ICNTC
      RETURN
42    PRINT 46
      STOP
43    PRINT 47
      STOP
44    PRINT 48, J
      STOP
```

```
C
45     FORMAT (1X,27HRETARDED TIMES FROM SEGMENT,I5,12H TO SEGMENTS,I5,
     1  4H AND,I5,10H DIFFER BY,I5,11H TIME STEPS)
46     FORMAT (1X,29HOVERFLOW IN FILLING ARRAY SHT)
47     FORMAT (1X,29HOVERFLOW IN FILLING ARRAY COF)
48     FORMAT (1X,29HSEGMENT CONNECTION ERROR,  J=,I5)
49     FORMAT ( //,29H LENGTH OF COEFFICIENT ARRAY=,I10,/,1X,I10,1X,
     1 25HBLOCKS OUTPUT TO FILE 11,,1X,I10,1X,24HBLOCKS OUTPUT TO FILE 1
     22,/)
       END
```

```
      SUBROUTINE  CONECT
C
C     CONNECT SETS UP SEGMENT CONNECTION DATA IN ARRAYS ICON1 AND ICON2
C     BY SEARCHING FOR SEGMENT ENDS THAT ARE IN CONTACT.
C
      COMMON /DATA/ N,NP,X(200),Y(200),Z(200),SI(200),BI(200),ALP(200),
     1   BET(200),ICON1(200),ICON2(200),ITAG(200),IPX,IPY,IPZ
      DIMENSION X2(1), Y2(1), Z2(1)
      EQUIVALENCE (X2(1),SI(1)), (Y2(1),ALP(1)), (Z2(1),BET(1))
      SMIN=1.E-3
      JNO=0
      DO 1 I=1,N
      ICON1(I)=0
1     ICON2(I)=0
      DO 33 I=1,N
      XI1=X(I)
      YI1=Y(I)
      ZI1=Z(I)
      XI2=X2(I)
      YI2=Y2(I)
      ZI2=Z2(I)
      SLEN=SQRT((XI2-XI1)**2+(YI2-YI1)**2+(ZI2-ZI1)**2)
C
C     DETERMINE CONNECTION DATA FOR END 1 OF SEGMENT.
C
      IF (IPX.EQ.0) GO TO 3
      SEP=XI1/SLEN
      SEP2=XI2/SLEN
      IF (SEP.GT.SMIN) GO TO 2
      IF (SEP.LT.-SMIN) GO TO 34
      IF (SEP2.LE.SMIN) GO TO 34
      ICON1(I)=20000+IPX
      GO TO 20
2     IF (SEP2.GT.SMIN) GO TO 3
      IF (SEP2.LT.-SMIN) GO TO 34
      ICON2(I)=20000+IPX
      GO TO 7
3     IF (IPY.EQ.0) GO TO 5
      SEP=YI1/SLEN
      SEP2=YI2/SLEN
      IF (SEP.GT.SMIN) GO TO 4
      IF (SEP.LT.-SMIN) GO TO 34
      IF (SEP2.LE.SMIN) GO TO 34
      ICON1(I)=20000+IPY
      GO TO 20
4     IF (SEP2.GT.SMIN) GO TO 5
      IF (SEP2.LT.-SMIN) GO TO 34
      ICON2(I)=20000+IPY
      GO TO 7
5     IF (IPZ.EQ.0) GO TO 7
      SEP=ZI1/SLEN
      SEP2=ZI2/SLEN
      IF (SEP.GT.SMIN) GO TO 6
      IF (SEP.LT.-SMIN) GO TO 34
      IF (SEP2.LE.SMIN) GO TO 34
      ICON1(I)=20000+IPZ
```

```
      GO TO 20
6     IF (SEP2.GT.SMIN) GO TO 7
      IF (SEP2.LT.-SMIN) GO TO 34
      ICON2(I)=20000+IPZ
7     IF (ICON1(I).NE.0) GO TO 20
      DO 9 IC=1,N
      IF (IC.EQ.I) GO TO 9
      ISEG=IC
      IF (ICON1(IC).NE.0) GO TO 8
      SEP=(ABS(XI1-X(IC))+ABS(YI1-Y(IC))+ABS(ZI1-Z(IC)))/SLEN
      IF (SEP.LT.SMIN) GO TO 10
8     IF (ICON2(IC).NE.0) GO TO 9
      SEP=(ABS(XI1-X2(IC))+ABS(YI1-Y2(IC))+ABS(ZI1-Z2(IC)))/SLEN
      IF (SEP.LT.SMIN) GO TO 15
9     CONTINUE
      GO TO 20
10    IF (ICON1(ISEG)) 12,11,13
11    ICON1(I)=ISEG
      ICON1(ISEG)=I
      GO TO 20
12    ICON1(I)=ICON1(ISEG)
      GO TO 20
13    JNO=JNO-1
      ICON1(I)=JNO
      IX=ICON1(ISEG)
      ICON1(ISEG)=JNO
      IF (ICON1(IX).EQ.ISEG) GO TO 14
      ICON2(IX)=JNO
      GO TO 20
14    ICON1(IX)=JNO
      GO TO 20
15    IF (ICON2(ISEG)) 17,16,18
16    ICON1(I)=ISEG
      ICON2(ISEG)=I
      GO TO 20
17    ICON1(I)=ICON2(ISEG)
      GO TO 20
18    JNO=JNO-1
      ICON1(I)=JNO
      IX=ICON2(ISEG)
      ICON2(ISEG)=JNO
      IF (ICON1(IX).EQ.ISEG) GO TO 19
      ICON2(IX)=JNO
      GO TO 20
19    ICON1(IX)=JNO
C
C     DETERMINE CONNECTION DATA FOR END 2 OF SEGMENT.
C
20    IF (ICON2(I).NE.0) GO TO 33
      DO 22 IC=1,N
      IF (IC.EQ.I) GO TO 22
      ISEG=IC
      IF (ICON1(IC).NE.0) GO TO 21
      SEP=(ABS(XI2-X(IC))+ABS(YI2-Y(IC))+ABS(ZI2-Z(IC)))/SLEN
      IF (SEP.LT.SMIN) GO TO 23
21    IF (ICON2(IC).NE.0) GO TO 22
      SEP=(ABS(XI2-X2(IC))+ABS(YI2-Y2(IC))+ABS(ZI2-Z2(IC)))/SLEN
      IF (SEP.LT.SMIN) GO TO 28
22    CONTINUE
      GO TO 33
```

```
23    IF (ICON1(ISEG)) 25,24,26
24    ICON2(I)=ISEG
      ICON1(ISEG)=I
      GO TO 33
25    ICON2(I)=ICON1(ISEG)
      GO TO 33
26    JNO=JNO-1
      ICON2(I)=JNO
      IX=ICON1(ISEG)
      ICON1(ISEG)=JNO
      IF (ICON1(IX).EQ.ISEG) GO TO 27
      ICON2(IX)=JNO
      GO TO 33
27    ICON1(IX)=JNO
      GO TO 33
28    IF (ICON2(ISEG)) 30,29,31
29    ICON2(I)=ISEG
      ICON2(ISEG)=I
      GO TO 33
30    ICON2(I)=ICON2(ISEG)
      GO TO 33
31    JNO=JNO-1
      ICON2(I)=JNO
      IX=ICON2(ISEG)
      ICON2(ISEG)=JNO
      IF (ICON1(IX).EQ.ISEG) GO TO 32
      ICON2(IX)=JNO
      GO TO 33
32    ICON1(IX)=JNO
33    CONTINUE
      RETURN
34    PRINT 35, I
      STOP
C
35    FORMAT (1X,7HSEGMENT,I4,33H LIES IN OR BEHIND SYMMETRY PLANE)
      END
```

```
      SUBROUTINE  CONSET
C
C     CONSET COMPUTES INTERPOLATION CONSTANTS FOR USE IN EVALUATING
C     THE FIELDS
C
      COMMON /DATA/ N,NP,X(200),Y(200),Z(200),SI(200),BI(200),ALP(200),
     1   BET(200),ICON1(200),ICON2(200),ITAG(200),IPX,IPY,IPZ
      COMMON /INTERP/ AT(3,200),BT(3,200),CT(3,200),ES(3),FS(3),GS(3),
     1   E(3),H(3)
      COMMON /ARRAX/ QI1,QI2,QI3,EC(200,5),EQ(200,5),IRET(200),BFR(15400
     1)
      COMMON /SCOMP/ SX(200),SY(200),SZ(200)
      COMMON /CONST/ CDT,VEL,DT,NTSTEP
      DO 5 I=1,N
      CALF=COS(ALP(I))
      SX(I)=CALF*COS(BET(I))
      SY(I)=CALF*SIN(BET(I))
      SZ(I)=SIN(ALP(I))
      IM=ICON1(I)
      IP=ICON2(I)
      IF (IM.GT.19000) IM=I
      IF (IP.GT.19000) IP=I
      S=SI(I)
      IF (IM.EQ.0) GO TO 1
      DM=.5*(S+SI(IM))
      GO TO 2
1     DM=.5*S
2     IF (IP.EQ.0) GO TO 3
      DP=.5*(S+SI(IP))
      GO TO 4
3     DP=.5*S
4     D1=DM*(DM+DP)
      D2=-DM*DP
      D3=DP*(DM+DP)
      AT(1,I)=1./D1
      AT(2,I)=1./D2
      AT(3,I)=1./D3
      BT(1,I)=-DP/D1
      BT(2,I)=(DM-DP)/D2
      BT(3,I)=DM/D3
      CT(1,I)=0.
      CT(2,I)=-DP*DM/D2
5     CT(3,I)=0.
      ES(1)=1./(2.*DT*DT)
      ES(2)=-2.*ES(1)
      ES(3)=ES(1)
      FS(3)=.5/DT
      FS(1)=-FS(3)
      FS(2)=0.
      GS(1)=0.
      GS(2)=1.
      GS(3)=0.
      C2=1./(VEL*VEL)
      E(1)=ES(1)*C2
      E(2)=ES(2)*C2
      E(3)=ES(3)*C2
```

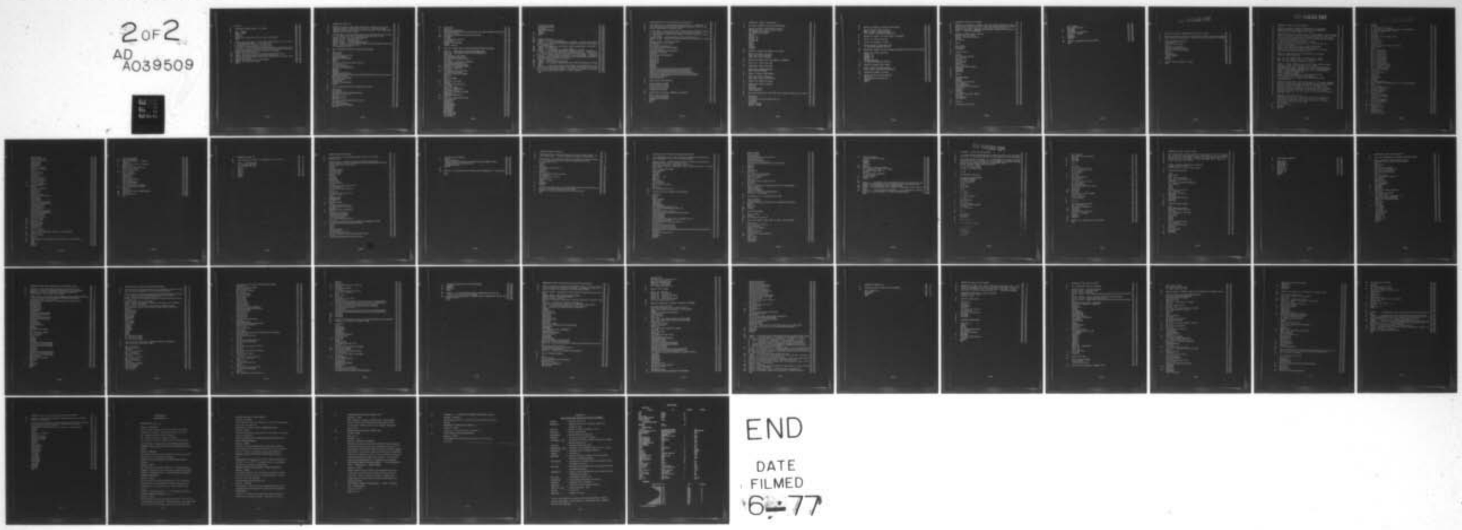
```
C2=-2./VEL
H(1)=ES(1)*C2
H(2)=ES(2)*C2
H(3)=ES(3)*C2
QI1=((ES(1)*DT/3.+.5*FS(1))*DT+GS(1))*DT
QI2=((ES(2)*DT/3.+.5*FS(2))*DT+GS(2))*DT
QI3=((ES(3)*DT/3.+.5*FS(3))*DT+GS(3))*DT
RETURN
END
```

```
      SUBROUTINE   CONVRT
C
C     CONVERT CHANGES GEOMETRY DATA FROM THE FORM STATING X,Y,Z
C     COORDINATES OF EACH SEGMENT END TO X,Y,Z OF THE SEGMENT CENTER
C     PLUS SEGMENT LENGTH AND ORIENTATION ANGLES AS REQUIRED IN MAIN
C     PROGRAM.
C
      COMMON /DATA/ N,NP,X(200),Y(200),Z(200),SI(200),BI(200),ALP(200),
     1   BET(200),ICON1(200),ICON2(200),ITAG(200),IPX,IPY,IPZ
      DIMENSION X2(1), Y2(1), Z2(1)
      EQUIVALENCE (X2(1),SI(1)), (Y2(1),ALP(1)), (Z2(1),BET(1))
      DO 1 I=1,N
      XA=X(I)
      YA=Y(I)
      ZA=Z(I)
      XB=X2(I)
      YB=Y2(I)
      ZB=Z2(I)
      X(I)=(XA+XB)*.5
      Y(I)=(YA+YB)*.5
      Z(I)=(ZA+ZB)*.5
      XA=XB-XA
      YA=YB-YA
      ZA=ZB-ZA
      SI(I)=SQRT(XA*XA+YA*YA+ZA*ZA)
      ALP(I)=ASIN(ZA/SI(I))
1     BET(I)=ATAN2(YA,XA)
      RETURN
      END
```

```
      SUBROUTINE  DATAGN
C
C     DATAGN IS THE MAIN ROUTINE FOR INPUT OF GEOMETRY DATA.
C
      COMMON /DATA/ N,NP,X(200),Y(200),Z(200),SI(200),BI(200),ALP(200),
     1 BET(200),ICON1(200),ICON2(200),ITAG(200),IPX,IPY,IPZ
      DIMENSION X2(1), Y2(1), Z2(1)
      DIMENSION ATST(4)
      INTEGER  GM,ATST
      EQUIVALENCE (X2(1),SI(1)), (Y2(1),ALP(1)), (Z2(1),BET(1))
      DATA  ATST/2HGW,2HGS,2HGE,2HGM/
      DATA  TA/.017453292/
      NWIRE=0
      N=0
      PRINT 8
      PRINT 9
C
C     READ GEOMETRY DATA CARD AND BRANCH TO SECTION FOR OPERATION
C     REQUESTED
C
1     READ 10, GM,ITG,NS,XW1,YW1,ZW1,XW2,YW2,ZW2,RAD
      IF (GM.EQ.ATST(1)) GO TO 2
      IF (GM.EQ.ATST(2)) GO TO 3
      IF (GM.EQ.ATST(3)) GO TO 6
      IF (GM.EQ.ATST(4)) GO TO 5
      GO TO 7
C
C     GENERATE SEGMENT DATA FOR STRAIGHT WIRE.
C
2     NWIRE=NWIRE+1
      I1=N+1
      I2=N+NS
      PRINT 11, NWIRE,XW1,YW1,ZW1,XW2,YW2,ZW2,RAD,NS,I1,I2,ITG
      CALL WIRE (XW1,YW1,ZW1,XW2,YW2,ZW2,RAD,NS,ITG)
      GO TO 1
C
C     SCALE STRUCTURE DIMENSIONS BY FACTOR XW1.
C
3     DO 4 I=1,N
      X(I)=X(I)*XW1
      Y(I)=Y(I)*XW1
      Z(I)=Z(I)*XW1
      X2(I)=X2(I)*XW1
      Y2(I)=Y2(I)*XW1
      Z2(I)=Z2(I)*XW1
4     BI(I)=BI(I)*XW1
      PRINT 12, XW1
      GO TO 1
C
C     MOVE STRUCTURE OR REPRODUCE ORIGINAL STRUCTURE IN NEW POSITIONS.
C
5     PRINT 13, ITG,NS,XW1,YW1,ZW1,XW2,YW2,ZW2,RAD
      XW1=XW1*TA
      YW1=YW1*TA
      ZW1=ZW1*TA
      CALL MOVE (XW1,YW1,ZW1,XW2,YW2,ZW2,INT(RAD+.5),NS,ITG)
```

AD-A039 509 ROME AIR DEVELOPMENT CENTER GRIFFISS AFB N Y F/G 9/5
A TIME DOMAIN PROGRAM FOR WIRE ANTENNA ANALYSIS.(U)
APR 77 J POTENZA
UNCLASSIFIED RADC-TR-77-123 NL

2 OF 2
AD
A039509



END
DATE
FILMED
6-77

```
      GO TO 1
C
C     TERMINATE STRUCTURE GEOMETRY INPUT.
C
6     CALL  CONECT
      CALL  CONVRT
      RETURN
7     PRINT 14
      PRINT 15, GM,ITG,NS,XW1,YW1,ZW1,XW2,YW2,ZW2,RAD
      STOP
C
8     FORMAT (////,33X,35H- - - STRUCTURE SPECIFICATION - - -,//,37X,
     1  28HCOORDINATES MUST BE INPUT IN,/,37X,
     2  29HMETERS OR BE SCALED TO METERS,/,37X,
     3  31HBEFORE STRUCTURE INPUT IS ENDED,//)
9     FORMAT (    2X,4HWIRE,79X,6HNO. OF,4X,5HFIRST,2X,4HLAST,5X,3HTAG,
     1 /,2X,3HNO.,8X,2HX1,9X,2HY1,9X,2HZ1,10X,2HX2,9X,2HY2,9X,2HZ2,6X,
     2 6HRADIUS,3X,4HSEG.,5X,4HSEG.,3X,4HSEG.,5X,3HNO.)
10    FORMAT (A2,I3,I5,7F10.5)
11    FORMAT (1X,I5,3F11.5,1X,4F11.5,2X,I5,4X,I5,1X,I5,3X,I5)
12    FORMAT (6X,26HSTRUCTURE SCALED BY FACTOR,F10.5)
13    FORMAT (6X,49HTHE STRUCTURE HAS BEEN MOVED, MOVE DATA CARD IS -/
     1 6X,I3,I5,7F10.5)
14    FORMAT (25H GEOMETRY DATA CARD ERROR)
15    FORMAT (1X,A2,I3,I5,7F10.5)
      END
```

```
      SUBROUTINE EINCF (T)
C
C     EINCF IS CALLED AT EACH TIME STEP TO FILL ARRAY EINC WITH THE
C     APPLIED FIELD ON EACH SEGMENT AT TIME T.  THIS ROUTINE USES A
C     GAUSSIAN PULSE OF FORM EXP(-(A**2)*(T-TMAX)**2)  OR GENERAL TIME
C     DEPENDENCE FROM INPUT TABLE)
C
      COMMON /DATA/  N,NP,X(200),Y(200),Z(200),SI(200),BI(200),ALP(200),
     1    BET(200),ICON1(200),ICON2(200),ITAG(200),IPX,IPY,IPZ
      COMMON /SCOMP/  SX(200),SY(200),SZ(200)
      COMMON /CONST/  CDT,VEL,DT,NTSTEP
      COMMON /EINC/  EINC(200),ESRC
      COMMON /ASORC/  EMAG(10),ISRC(10),NSRC
      COMMON /ENSET/  A,TMAX,EIX,PX,PY,PZ,EX,EY,EZ,VLT(200),TW,IARB,TAMX
      DATA    TA/0.017453292/,XW/0.1/
      IF (NSRC.EQ.0) GO TO 8
C
C     SET UP APPLIED FIELD FOR TRANSMITTING ANTENNA
C
      DO 1 I=1,N
    1 EINC(I)=0.
      IF (IARB.NE.0) GO TO 2
      ESRC=A*(T-TMAX)
      ESRC=EXP(-ESRC*ESRC)
      GO TO 6
    2 TD=T
    3 IF (TD.LT.0..OR.TD.GT.TAMX) GO TO 4
      IS=TD/TW+1.5
      IF (IS.LT.2) IS=2
      IF (IS.EQ.IARB) IS=IS-1
      GO TO 5
    4 ESRC=0.
      RETURN
    5 TD=TD-TW*FLOAT(IS-1)
      ESRC=(.5*TD*((TD-TW)*VLT(IS-1)+(TD+TW)*VLT(IS+1))-(TD+TW)*(TD-TW)*
     1VLT(IS))/(TW*TW)
      IF (NSRC.EQ.0) RETURN
    6 DO 7 IS=1,NSRC
      I=ISRC(IS)
    7 EINC(I)=EMAG(IS)*ESRC
      RETURN
C
C     SET UP APPLIED FIELD FOR INCIDENT PLANE WAVE
C
    8 TCON=T-EIX
      DO 13 I=1,N
      TD=TCON-(X(I)*PX+Y(I)*PY+Z(I)*PZ)
      IF (IARB.NE.0) GO TO 9
      TD=TD*A
      ESRC=EXP(-TD*TD)
      GO TO 12
    9 IF (TD.LT.0..OR.TD.GT.TAMX) GO TO 10
      IS=TD/TW+1.5
      IF (IS.LT.2) IS=2
      IF (IS.EQ.IARB) IS=IS-1
      GO TO 11
```

```
10    EINC(I)=0.
      GO TO 13
11    TD=TD-TW*FLOAT(IS-1)
      ESRC=(.5*TD*((TD-TW)*VLT(IS-1)+(TD+TW)*VLT(IS+1))-(TD+TW)*(TD-TW)*
     1VLT(IS))/(TW*TW)
12    EINC(I)=ESRC*(SX(I)*EX+SY(I)*EY+SZ(I)*EZ)
13    CONTINUE
      TD=TCON
      IF (IARB.NE.0) GO TO 3
      ESRC=A*TCON
      ESRC=EXP(-ESRC*ESRC)
      RETURN
      ENTRY   ESET(T)
C
C     READ DATA CARDS AND INITIALIZE CONSTANTS FOR EXCITATION
C
      READ 25,  IARB,NSRC,TW,THET,PHI,ET,RZERO,TMAX
      PRINT 31,  IARB,NSRC,TW,THET,PHI,ET,RZERO,TMAX
      PRINT 26
      IF (IARB.NE.0) GO TO 15
      A=SQRT(-ALOG(XW))/(TW*.5)
      IF (TMAX.GT.1.E-20) GO TO 14
      EIN=0.03
      IF (TMAX.LT.(-1.E-20)) EIN=-TMAX
      TMAX=SQRT(-ALOG(EIN))/A
      TMAX=(AINT(TMAX/DT)+1.)*DT
14    EIX=EXP(-A*A*TMAX*TMAX)*100.
      PRINT 27, A,TMAX,EIX
      GO TO 17
15    IF (IARB.GT.200) GO TO 20
      TAMX=TW*FLOAT(IARB-1)
      READ 21, (VLT(I),I=1,IARB)
      PRINT 23
      T=-TW
      DO 16 I=1,IARB
      T=T+TW
16    PRINT 22, I,T,VLT(I)
      TMAX=0.
17    IF (NSRC.EQ.0) GO TO 19
      PRINT 28
      DO 18 IS=1,NSRC
      READ 29,  ITG,II,VMAG
      I=ISEGNO(ITG,II)
      PRINT 30,  IS,ITG,II,I,VMAG
      ISRC(IS)=I
18    EMAG(IS)=VMAG/SI(I)
      RETURN
19    PRINT 32, RZERO,THET,PHI,ET
      THET=THET*TA
      PHI=PHI*TA
      ET=ET*TA
      ST=SIN(THET)
      CT=COS(THET)
      SP=SIN(PHI)
      CP=COS(PHI)
      SE=SIN(ET)
      CE=COS(ET)
      PX=-ST*CP/VEL
      PY=-ST*SP/VEL
      PZ=-CT/VEL
```

```
      EX=CT*CP*CE-SP*SE
      EY=CT*SP*CE+CP*SE
      EZ=-ST*CE
      EIX=TMAX+RZERO/VEL
      RETURN
20    PRINT 24
      STOP
C
C
21     FORMAT (6E12.5)
22     FORMAT (30X,I5,E13.5,E14.5)
23     FORMAT (    31X,52HTIME DEPENCENCE - GENERAL,  SET BY INPUT TABLE B
      1ELOW.//,31X,4HSTEP,7X,4HTIME,9X, 6HSOURCE,/,32X,3HNO.,6X,6H(SEC.),
      2  7X,8HSTRENGTH)
24     FORMAT (52H NUMBER OF EXCITATION VALUES EXCEEDS ARRAY DIMENSION)
25     FORMAT  (2I5,6E10.3)
26     FORMAT  ( //,38X,26H- - - - EXCITATION - - - -,//)
27     FORMAT (  31X,78HTIME DEPENDENCE- GAUSSIAN, PEAK AT TIME=TMAX - -
      1  EXP(-(A**2)*(TIME-TMAX)**2),//,31X,2HA=,E12.5,9H,    TMAX=,E12.5,
      2 //,31X,10HSOURCE HAS,F8.3,33H PERCENT OF PEAK VALUE AT TIME=0.)
28     FORMAT ( //,31X,17HVOLTAGE SOURCES -,//,31X.23HSOURCE    SOURCE SEG
      1MENT,6X,12HPEAK VOLTAGE,/,33X,3HNO.,4X,3HTAG,2X,4HINC.,2X,4HSEG.,
      2 8X,7H(VOLTS),/,40X,3HNO.,9X,3HNO.)
29     FORMAT  (2I5,E10.3)
30     FORMAT  (31X,I5,I7,2I6,5X,F10.4)
31     FORMAT  ( //,1X,26H*** EXCITATION DATA CARD**,2I5,E12.5,4F10.5,E12
      1.5)
32     FORMAT (  //,31X,22HINCIDENT PLANE PULSE -,/,31X,65HAT TIME = 0. L
      1EADING EDGE OF PULSE IS SHIFTED BACK FROM ORIGIN BY,F10.5,7H METER
      2S, //,31X,26HINCIDENCE ANGLES -  THETA=.F10.5.5H DEG.,/,51X,6HPHI
      3 =,F10.5,5H DEG.,/,51X,6HETA  =,F10.5,5H DEG.,/)
       END
```

```
      SUBROUTINE EMAT (RX,RY,RZ,R2,TAU,SXJ,SYJ,SZJ,J)
C
C     EMAT COMPUTES THE COEFFICIENTS WHICH MULTIPLY A 3 SEGMENT BY 3
C     TIME STEP PATCH OF CURRENT AND INTEGRAL OF CURRENT TO YIELD THE
C     VECTOR COMPONENTS OF ELECTRIC FIELD
C
      COMMON /DATA/  N,NP,X(200),Y(200),Z(200),SI(200),BI(200),ALP(200),
     1    BET(200),ICON1(200),ICON2(200),ITAG(200),IPX,IPY,IPZ
      COMMON /INTERP/  AT(3,200),BT(3,200),CT(3,200),ES(3),FS(3),GS(3),
     1    E(3),H(3)
      COMMON /INTG/  X1,X2,X3,X4,X5,X6,X7,X8,X9,X10,X11,X12
      COMMON /EMATS/  EMX(3,3),EMY(3,3),EMZ(3,3),QMX(3,3),QMY(3,3),QMZ(3
     1,3)
      COMMON /CONST/  CDT,VEL,DT,NTSTEP
      DIMENSION P(3), G(3), F(3)
      DO 1 L=1,3
      P(L)=2.*ES(L)*TAU+FS(L)
      G(L)=(ES(L)*TAU+FS(L))*TAU+GS(L)
    1 F(L)=-P(L)/VEL
      BCON=-2.*(RX*SXJ+RY*SYJ+RZ*SZJ)
      CCON=R2+BI(J)*BI(J)
      SJ=SI(J)
      CALL INTEG (SJ,BCON,CCON)
      DO 2 L=1,3
      AL=AT(L,J)
      BL=BT(L,J)
      CL=CT(L,J)
      DO 2 M=1,3
      EM=E(M)
      FM=F(M)
      GM=G(M)
      T1=H(M)*(AL*X9+CL*X1)+P(M)*(AL*X10+BL*X6+CL*X2)
      T2=2.*AL*(FM*X6+GM*X7)+BL*(EM*X1+FM*X2+GM*X3)
      T3=2.*AL*(EM*X9+FM*X10+GM*X11)+BL*(FM*X6+GM*X7)
      T4=2.*AL*(EM*X6+FM*X7+GM*X8)+BL*(EM*X2+FM*X3+GM*X4)
      T5=2.*AL*(EM*X10+FM*X11+GM*X12)+BL*(EM*X6+FM*X7+GM*X8)
      T1=-(T1-VEL*T3)*1.E-7
      T2=-VEL*T2*1.E-7
C
C     COEFFICIENTS FOR CURRENTS
C
      EMX(L,M)=SXJ*T1+RX*T2
      EMY(L,M)=SYJ*T1+RY*T2
      EMZ(L,M)=SZJ*T1+RZ*T2
      T4=-VEL*VEL*T4*1.E-7
      T5=VEL*VEL*T5*1.E-7
C
C     COEFFICIENTS FOR TIME INTEGRAL OF CURRENTS
C
      QMX(L,M)=RX*T4+SXJ*T5
      QMY(L,M)=RY*T4+SYJ*T5
    2 QMZ(L,M)=RZ*T4+SZJ*T5
      RETURN
      END
```

```
      SUBROUTINE FACIO (A,AR,N,IX,IP)
C
C     S/R WHICH CONTROLS I/O FOR FACTORIZATION
C
      COMMON /MATPAR/ NBLOKS,NPBLK,NLAST,INT
      DIMENSION A(N,N), IP(N), IX(N), AR(1)
      IF (NBLOKS.GT.2) GO TO 1
      CALL FACTR (N,A,IP,N)
      RETURN
1     REWIND  11
      REWIND  13
      REWIND  14
      IT=NPBLK*N
      I1=1
      ITWO=2
      I2=IT
      I3=I2+1
      I4=2*IT
C
C     BUFFER IN BLOCK1 AND BLOCK2 FROM TAPE 1
C
      READ (13) (AR(I),I=I1,I2)
      READ (13) (AR(I),I=I3,I4)
      CALL LFACTR (A,N,1,2,IP)
C
C     BUFFER OUT BLOCK1 TO TAPE2 (BLOCK1 FACTORED)
C
      WRITE (11) (AR(I),I=I1,I2)
C
C     BUFFER OUT BLOCK2 TO FILE3
C
      WRITE (14) (AR(I),I=I3,I4)
      DO 2 IXBLK2=3,NBLOKS
C
C     BUFFER IN BLOCK2 FROM TAPE1
C
      READ (13) (AR(I),I=I3,I4)
      CALL LFACTR (A,N,1,IXBLK2,IP)
C
C     BUFFER OUT BLOCK2 TO FILE3
C
      WRITE (14) (AR(I),I=I3,I4)
2     CONTINUE
      IXBLK1=1
3     IXBLK1=IXBLK1+1
      IXBLK2=IXBLK1+1
C
C     WITH THE EXCEPTION OF THE FIRST PASS, IFILE3 BECOMES IFILE4 AND VI
C
      IFILE3=13
      IFILE4=14
      IF (2*(IXBLK1/2).NE.IXBLK1) GO TO 4
      IFILE3=14
      IFILE4=13
4     REWIND  IFILE3
      REWIND  IFILE4
```

```
C
C     BUFFER IN BLOCK1 AND BLOCK2 FROM IFILE3
C
      READ (IFILE3) (AR(I),I=I1,I2)
      READ (IFILE3) (AR(I),I=I3,I4)
      CALL LFACTR (A,N,IXBLK1,IXBLK2,IP)
C
C     BUFFER OUT BLOCK1 TO TAPE2 (BLOCK1 FACTORED)
C
      WRITE (11) (AR(I),I=I1,I2)
C
C     BUFFER OUT BLOCK2 TO IFILE4
C
      WRITE (IFILE4) (AR(I),I=I3,I4)
      IF (IXBLK2.NE.NBLOKS) GO TO 5
C
C     BUFFER OUT BLOCK2 TO TAPE2 (BLOCK1 FACTORED--FACTORIZATION FINISHE
C
      WRITE (11) (AR(I),I=I3,I4)
      REWIND  11
      REWIND  13
      REWIND  14
      GO TO 6
5     IXBLK2=IXBLK2+1
      IF (IXBLK2.GT.NBLOKS) GO TO 3
C
C     BUFFER IN BLOCK2 FROM IFILE3
C
      READ (IFILE3) (AR(I),I=I3,I4)
      CALL LFACTR (A,N,IXBLK1,IXBLK2,IP)
C
C     BUFFER OUT BLOCK2 TO FILE4
C
      WRITE (IFILE4) (AR(I),I=I3,I4)
      GO TO 5
6     CALL LUNSCR (A,AR,N,IX,IP)
      RETURN
      END
```

```
      SUBROUTINE FACTR (N,A,P,NDIM)
C
C     SUBROUTINE TO FACTOR A MATRIX INTO A UNIT LOWER TRIANGULAR MATRIX
C     AND AN UPPER TRIANGULAR MATRIX USING THE GAUSS-DOOLITTLE ALGORITHM
C     PRESENTED ON PAGES 411-416 OF A. RALSTON--A FIRST COURSE IN
C     NUMERICAL ANALYSIS.  COMMENTS BELOW REFER TO COMMENTS IN RALSTONS
C     TEST.     (MATRIX TRANSPOSED)
C
      DIMENSION A(NDIM,NDIM), P(NDIM)
      COMMON /SCRATM/ D(200)
      INTEGER  R,P,RM1,RP1,PJ,PR
      IFLG=0
      DO 9 R=1,N
C
C     STEP 1
C
      DO 1 K=1,N
      D(K)=A(R,K)
1     CONTINUE
C
C     STEPS 2 AND 3
C
      RM1=R-1
      IF (RM1.LT.1) GO TO 4
      DO 3 J=1,RM1
      PJ=P(J)
      A(R,J)=D(PJ)
      D(PJ)=D(J)
      JP1=J+1
      DO 2 I=JP1,N
      D(I)=D(I)-A(J,I)*A(R,J)
2     CONTINUE
3     CONTINUE
4     CONTINUE
C
C     STEP 4
C
      DMAX=ABS(D(R))
      P(R)=R
      RP1=R+1
      IF (RP1.GT.N) GO TO 6
      DO 5 I=RP1,N
      ELMAG=ABS(D(I))
      IF (ELMAG.LT.DMAX) GO TO 5
      DMAX=ELMAG
      P(R)=I
5     CONTINUE
6     CONTINUE
      IF (DMAX.LT.1.E-10) IFLG=1
      PR=P(R)
      A(R,R)=D(PR)
      D(PR)=D(R)
C
C     STEP 5
C
      IF (RP1.GT.N) GO TO 8
```

```
      DO 7 I=RP1,N
      A(R,I)=D(I)/A(R,R)
7     CONTINUE
8     CONTINUE
      IF (IFLG.EQ.0) GO TO 9
      PRINT 10,  R,DMAX
      IFLG=0
9     CONTINUE
      RETURN
C
10    FORMAT (1X,6HPIVOT(,I3,2H)=,E16.8)
      END
```

```
      SUBROUTINE FBLOCK (NBLOKS,NPBLK,NLAST,IMAX,N,INT)
C
C     FBLOCK DETERMINES BLOCK SIZE AND NUMBER OF BLOCKS WHEN OUT-OF-CORE
C     MATRIX STORAGE IS REQUIRED.  INT=0 RETURNED IF MATRIX FITS IN CORE
C
      IF (N*N.LE.IMAX) GO TO 1
      NPBLK=IMAX/(2*N)
      IF (NPBLK.LT.1) STOP
      INT=1
      NBLOKS=(N-1)/NPBLK+1
      NLAST=N-(NBLOKS-1)*NPBLK
      PRINT 2, NBLOKS,NPBLK,NLAST
      RETURN
1     NBLOKS=2
      NPBLK=(N+1)/2
      NLAST=N-NPBLK
      INT=0
      RETURN
C
2     FORMAT (1X,11H BLOCKING   ,4I5/)
      END
```

```
      SUBROUTINE FORT (A,M,S,IFS,IFERR)
C
C     FOURIER TRANSFORM SUBROUTINE, PROGRAMMED IN SYSTEM/360,
C     BASIC PROGRAMMING SUPPORT, FORTRAN IV, FORM C28-6504
C     THIS DECK SET UP FOR IBSYS ON IBM 7094,
C
C     DOES EITHER FOURIER SYNTHESIS,I.E.,COMPUTES COMPLEX FOURIER SERIES
C     GIVEN A VECTOR OF N COMPLEX FOURIER AMPLITUDES,OR, GIVEN A VECTOR
C     OF COMPLEX DATA X DOES FOURIER ANALYSIS, COMPUTING AMPLITUDES.
C     A IS A COMPLEX VECTOR OF LENGTH N=2**M COMPLEX NOS. OR 2*N REAL
C     NUMBERS. A IS TO BE SET BY USER.
C     M  IS AN INTEGER 0.LT.M.LE.13, SET BY USER.
C     S IS A VECTOR S(J)= SIN(2*PI*J/NP ), J=1,2,....,NP/4-1,
C     COMPUTED BY PROGRAM.
C     IFS IS A PARAMETER TO BE SET BY USER AS FOLLOWS-
C     IFS=0 TO SET NP=2**M AND SET UP SINE TABLE.
C     IFS=1 TO SET N=NP=2**M, SET UP SIN TABLE, AND DO FOURIER
C     SYNTHESIS, REPLACING THE VECTOR A BY
C
C     X(J)= SUM OVER K=0,N-1 OF A(K)*EXP(2*PI*I/N)**(J*K),
C     J=0,N-1, WHERE I=SQRT(-1)
C
C     THE  X'S  ARE STORED WITH   RE X(J) IN CELL  2*J+1
C     AND  IM X(J)  IN CELL  2*J+2  FOR  J=0,1,2,....,N-1.
C     THE  A'S  ARE STORED IN THE SAME MANNER.
C
C     IFS=-1   TO SET  N=NP=2**M,SET UP SIN TABLE. AND DO FOURIER
C     ANALYSIS, TAKING THE INPUT VECTOR A AS X AND
C     REPLACING IT BY THE A  SATISFYING THE ABOVE FOURIER SERIES.
C     IFS=+2 TO DO FOURIER SYNTHESIS ONLY, WITH A PRE-COMPUTED S.
C     IFS=-2 TO DO FOURIER ANALYSIS ONLY, WITH A PRE-COMPUTED S.
C     IFERR   IS SET BY PROGRAM TO-
C     =0 IF NO ERROR DETECTED.
C     =1 IF M IS OUT OF RANGE., OR, WHEN IFS=+2,-2, THE
C     PRE-COMPUTED S TABLE IS NOT LARGE ENOUGH.
C     =-1 WHEN IFS =+1,-1, MEANS ONE IS RECOMPUTING S TABLE
C     UNNECESSARILY.
C
C     NOTE- AS STATED ABOVE, THE MAXIMUM VALUE OF M FOR THIS PROGRAM
C     ON THE IBM 7094 IS 13. FOR 360 MACHINES HAVING GREATER STORAGE
C     CAPACITY, ONE MAY INCREASE THIS LIMIT BY REPLACING 13 IN
C     STATEMENT 3 BELOW BY LOG2 N, WHERE N IS THE MAX. NO. OF
C     COMPLEX NUMBERS ONE CAN STORE IN HIGH-SPEED CORE.  ONE MUST
C     ALSO ADD MORE  DO  STATEMENTS TO THE BINARY SORT ROUTINE
C     FOLLOWING STATEMENT  24  AND CHANGE THE EQUIVALENCE STATEMENTS
C     FOR THE  K'S.
C
      DIMENSION A(1), S(1), K(14)
      EQUIVALENCE (K(13),K1), (K(12),K2), (K(11),K3), (K(10),K4)
      EQUIVALENCE (K(9),K5), (K(8),K6), (K(7),K7), (K(6),K8)
      EQUIVALENCE (K(5),K9), (K(4),K10), (K(3),K11), (K(2),K12)
      EQUIVALENCE (K(1),K13), (K(1),N2)
      IF (M) 2,2,1
1     IF (M-13) 4,4,2
2     IFERR=1
3     RETURN
```

```
4       IFERR=0
        N=2**M
        IF (IABS(IFS)-1) 25,25,5
C       WE ARE DOING TRANSFORM ONLY. SEE IF PRE-COMPUTED
C       S TABLE IS SUFFICIENTLY LARGE
5       IF (N-NP) 7,7,6
6       IFERR=1
        GO TO 25
C       SCRAMBLE A, BY SANDE'S METHOD
7       K(1)=2*N
        DO 8 L=2,M
8       K(L)=K(L-1)/2
        DO 9 L=M,12
9       K(L+1)=2
C       NOTE EQUIVALENCE OF KL AND K(14-L)
C       BINARY SORT-
        IJ=2
        DO 11 J1=2,K1,2
        DO 11 J2=J1,K2,K1
        DO 11 J3=J2,K3,K2
        DO 11 J4=J3,K4,K3
        DO 11 J5=J4,K5,K4
        DO 11 J6=J5,K6,K5
        DO 11 J7=J6,K7,K6
        DO 11 J8=J7,K8,K7
        DO 11 J9=J8,K9,K8
        DO 11 J10=J9,K10,K9
        DO 11 J11=J10,K11,K10
        DO 11 J12=J11,K12,K11
        DO 11 JI=J12,K13,K12
        IF (IJ-JI) 10,11,11
10      T=A(IJ-1)
        A(IJ-1)=A(JI-1)
        A(JI-1)=T
        T=A(IJ)
        A(IJ)=A(JI)
        A(JI)=T
11      IJ=IJ+2
        IF (IFS) 12,2,14
C       DOING FOURIER ANALYSIS,SO DIV. BY N AND CONJUGATE.
12      FN=N
        DO 13 I=1,N
        A(2*I-1)=A(2*I-1)/FN
13      A(2*I)=-A(2*I)/FN
C       SPECIAL CASE- L=1
14      DO 15 I=1,N,2
        T=A(2*I-1)
        A(2*I-1)=T+A(2*I+1)
        A(2*I+1)=T-A(2*I+1)
        T=A(2*I)
        A(2*I)=T+A(2*I+2)
15      A(2*I+2)=T-A(2*I+2)
        IF (M-1) 2,3,16
C       SET FOR L=2
16      LEXP1=2
C       LEXP1=2**(L-1)
        LEXP=8
C       LEXP=2**(L+1)
        NPL=2**MT
C       NPL = NP* 2**-L
```

```
      DO 22 L=2,M
C     SPECIAL CASE- J=0
      DO 17 I=2,N2,LEXP
      I1=I+LEXP1
      I2=I1+LEXP1
      I3=I2+LEXP1
      T=A(I-1)
      A(I-1)=T+A(I2-1)
      A(I2-1)=T-A(I2-1)
      T=A(I)
      A(I)=T+A(I2)
      A(I2)=T-A(I2)
      T=-A(I3)
      TI=A(I3-1)
      A(I3-1)=A(I1-1)-T
      A(I3)=A(I1)-TI
      A(I1-1)=A(I1-1)+T
17    A(I1)=A(I1)+TI
      IF (L-2) 21,21,18
18    KLAST=N2-LEXP
      JJ=NPL
      DO 20 J=4,LEXP1,2
      NPJJ=NT-JJ
      UR=S(NPJJ)
      UI=S(JJ)
      ILAST=J+KLAST
      DO 19 I= J,ILAST,LEXP
      I1=I+LEXP1
      I2=I1+LEXP1
      I3=I2+LEXP1
      T=A(I2-1)*UR-A(I2)*UI
      TI=A(I2-1)*UI+A(I2)*UR
      A(I2-1)=A(I-1)-T
      A(I2)=A(I)-TI
      A(I-1)=A(I-1)+T
      A(I)=A(I)+TI
      T=-A(I3-1)*UI-A(I3)*UR
      TI=A(I3-1)*UR-A(I3)*UI
      A(I3-1)=A(I1-1)-T
      A(I3)=A(I1)-TI
      A(I1-1)=A(I1-1)+T
19    A(I1)=A(I1)+TI
C     END OF I LOOP
20    JJ=JJ+NPL
C     END OF J LOOP
21    LEXP1=2*LEXP1
      LEXP=2*LEXP
22    NPL=NPL/2
C     END OF L LOOP
      IF (IFS) 23,2,3
C     DOING FOURIER ANALYSIS. REPLACE A BY CONJUGATE.
23    DO 24 I=1,N
24    A(2*I)=-A(2*I)
      GO TO 3
C     RETURN
C     MAKE TABLE OF S(J)=SIN(2*PI*J/NP),J=1,2,....,NT-1,NT=NP/4
25    NP=N
      MP=M
      NT=N/4
      MT=M-2
```

```
      IF (MT) 31,31,26
26    THETA=.785398163
C     THETA=PI/2**(L+1)     FOR L=1
      JSTEP=NT
C     JSTEP = 2**( MT-L+1 ) FOR L=1
      JDIF=NT/2
C     JDIF = 2**(MT-L)  FOR L=1
      S(JDIF)=SIN(THETA)
      IF (MT-2) 31,27,27
27    DO 30 L=2,MT
      THETA=THETA/2.
      JSTEP2=JSTEP
      JSTEP=JDIF
      JDIF=JDIF/2
      S(JDIF)=SIN(THETA)
      JC1=NT-JDIF
      S(JC1)=COS(THETA)
      JLAST=NT-JSTEP2
      IF (JLAST-JSTEP) 30,28,28
28    DO 29 J=JSTEP,JLAST,JSTEP
      JC=NT-J
      JD=J+JDIF
29    S(JD)=S(J)*S(JC1)+S(JDIF)*S(JC)
30    CONTINUE
31    IF (IFS) 7,3,7
      END
```

```
      SUBROUTINE IFEH (I)
C
C     IFEH CONVERTS I FROM E, H, OR BLANK TO 1, -1, OR 0.
C
      DATA    IE,IH/1HE,1HH/
      IF (I.EQ.IE) GO TO 1
      IF (I.EQ.IH) GO TO 2
      I=0
      RETURN
1     I=1
      RETURN
2     I=-1
      RETURN
      END
```

```
      SUBROUTINE INTEG (EL,B,C)
C
C     INTEG COMPUTES 12 INTEGRALS NEEDED TO SET UP THE FIELD
C     COEFFICIENTS
C
      COMMON /INTG/  X1,X2,X3,X4,X5,X6,X7,X8,X9,X10,X11,X12
      DOUBLE PRECISION  S2,S1,R1S,R2S,R1,R2,B2,HB,DE1,DE2,DIS,ALR,DIH,
     1  DE1S,DE2S,T1,T2,D2,D3,D4,D8
      S2=EL*.5
      S1=-S2
      R1S=(S1+B)*S1+C
      R2S=(S2+B)*S2+C
      R1=DSQRT(R1S)
      R2=DSQRT(R2S)
      B2=B*B
      HB=.5*B
      DE1=S1+HB
      DE2=S2+HB
      DIS=4.*C-B2
      ALR=DLOG(R2/R1)
      LIM=0
      IF (ABS(B).LT.1.E-27) GO TO 2
      XLIM=DIS/ABS(B)
      IF (XLIM.GT.1.E-5) GO TO 2
      LIM=1
      DIH=0.D+0
      IF (DE1*DE2.GT.0.) GO TO 1
      PRINT 10,  B,C
      STOP
1     DE1S=DE1*DE1
      DE2S=DE2*DE2
      GO TO 3
2     DIH=DSQRT(DIS)
3     X1=EL
      IF (DE1.LT.0.) GO TO 4
      D2=DLOG((R2+DE2)/(R1+DE1))
      GO TO 5
4     D2=DLOG((R1-DE1)/(R2-DE2))
5     X2=D2
      IF (LIM.EQ.1) GO TO 6
      T2=DATAN(2.*DE2/DIH)
      T1=DATAN(2.*DE1/DIH)
      D3=2.*(T2-T1)/DIH
      D4=4.*(DE2/R2-DE1/R1)/DIS
      GO TO 7
6     D3=(DIS/(12.*DE2*DE2)-1.)/DE2-(DIS/(12.*DE1*DE1)-1.)/DE1
      D4=.5*DABS((DE2/DE1-DE1/DE2)/(R1*R2))
7     X3=D3
      X4=D4
      X5=0.
      X6=R2-R1-HB*D2
      X7=ALR-HB*D3
      IF (LIM.EQ.1) GO TO 8
      D8=-2.*((B*S2+2.*C)/R2-(B*S1+2.*C)/R1)/DIS
      GO TO 9
8     D8=.25*(B/DE2S-B/DE1S)+1./DE1-1./DE2
```

```
      IF (DE1.LT.0.) D8=-D8
9     X8=D8
      X9=EL*EL*EL/12.
      X10=.5*((S2-1.5*B)*R2-(S1-1.5*B)*R1)+.125*(3.*B2-4.*C)*D2
      X11=S2-S1-B*ALK+.5*(B2-2.*C)*D3
      X12=D2-B*D8-C*D4
      RETURN
C
C
10    FORMAT (1X,45HINTEGRATION ATTEMPTED OVER SINGULARITY,  B,C=,2E15.6
     1)
      END
```

```
      FUNCTION ISEGNO (ITAGI,M)
C
C     ISEGNO RETURNS THE SEGMENT NUMBER OF THE MTH SEGMENT HAVING THE
C     TAG NUMBER ITAGI.  IF ITAGI=0 SEGMENT NUMBER M IS RETURNED.
C
      COMMON /DATA/ N,NP,X(200),Y(200),Z(200),SI(200),BI(200),ALP(200),
     1   BET(200),ICON1(200),ICON2(200),ITAG(200),IPX,IPY,IPZ
      IF (M.GT.0) GO TO 1
      PRINT 5
      STOP
1     ICNT=0
      IF (ITAGI.NE.0) GO TO 2
      ISEGNO=M
      RETURN
2     DO 3 I=1,N
      IF (ITAG(I).NE.ITAGI) GO TO 3
      ICNT=ICNT+1
      IF (ICNT.EQ.M) GO TO 4
3     CONTINUE
      PRINT 6, ITAGI
      STOP
4     ISEGNO=I
      RETURN
C
5     FORMAT (4X,91HCHECK DATA, PARAMETER SPECIFYING SEGMENT POSITION IN
     1 A GROUP OF EQUAL TAGS MUST NOT BE ZERO)
6     FORMAT (///,10X,26HNO SEGMENT HAS AN ITAG OF ,I5)
      END
```

```
      SUBROUTINE ITOF (TRAN,NT,M,NOE,NOC,NTNX,NTNR)
C
C     ITOF PROCESSES DATA IN ARRAY TRAN FOR FOURIER TRANSFORMING AND
C     CALLS SUBROUTINE FORT TO DO THE TRANSFORMING
C
      COMMON /ARRAX/  QI1,QI2,QI3,ET(200,5,8),EP(200,5,8),IRET(200,8)
      DIMENSION TRAN(1024), A(4096), S(511)
      DIMENSION TZERO(100), AMAX(100), TEXT(100)
      EQUIVALENCE (A,QI1), (TZERO,EP(1,1,2)), (AMAX,EP(1,1,3)), (TEXT,EP
     1(1,1,4))
      DATA    MMAX,ITZX/0,100/
      PI=3.1415926
      TP=2.*PI
      IFERR=0
      N=2**M
      PRINT 24,  NT,N
      N2=N*2
      NTP=NT+1
      DO 1 I=1,NT
    1 A(I)=TRAN(I)
      IF (NT.LT.N) GO TO 2
      NT=N
      GO TO 12
    2 IF (NOE.NE.0) GO TO 12
C
C     EXAMINE DATA FOR POSSIBLE EXTRAPOLATION
C
      NTM=NT-1
      ITZ=0
      ITM=0
      DO 4 I=3,NTM
      TRA=A(I-1)
      TRB=A(I)
      TRC=A(I+1)
      ATRA=ABS(TRA)
      IF (ATRA.LT.1.E-35) GO TO 3
      IF (1.E+10*TRA*TRB.GT.0.) GO TO 3
      ITZ=ITZ+1
      IF (ITZ.GT.ITZX) GO TO 11
      TZERO(ITZ)=FLOAT(I-2)+ABS(TRA/(TRA-TRB))
    3 ATRA=ABS(TRA)
      ATRB=ABS(TRB)
      ATRC=ABS(TRC)
      IF ((ATRB.LE.ATRA).OR.(ATRB.LT.ATRC)) GO TO 4
      TMAXI=TRA-2.*TRB+TRC
      IF (ABS(TMAXI).LT.1.E-35) GO TO 4
      ITM=ITM+1
      IF (ITM.GT.ITZX) GO TO 11
      TMAXI=-.5*(TRC-TRA)/TMAXI
      TEXT(ITM)=TMAXI+FLOAT(I-1)
      AMAX(ITM)=(.5*TRA-TRB+.5*TRC)*TMAXI*TMAXI+(TRC-TRA)*.5*TMAXI+TRB
    4 CONTINUE
      IF (ITM.LT.3) GO TO 11
      ALFA=0.
      ICOUNT=0
      ISSS=2
```

```
      AMAXL=AMAX(2)
      DO 7 I=5,ITM
      AMAXI=AMAX(I)
      IF (AMAXI*AMAXL.GE.0.) GO TO 5
      ICOUNT=ICOUNT+1
      AMAXR=-AMAXI/AMAXL
      IF (AMAXR.GT.1.) GO TO 5
      ALFA=ALFA+ALOG(AMAXR)/(TEXT(I-1)-TEXT(I))
      GO TO 6
5     ICOUNT=0
      ISSS=I
      ALFA=0.
6     AMAXL=AMAXI
7     CONTINUE
      IF (ICOUNT.LT.3) GO TO 11
      ALFA=ALFA/FLOAT(ICOUNT)
      TMAXI=TEXT(ISSS)
      DO 8 I=1,ITZ
      ICOUNT=I
      IF (TZERO(I).GT.TMAXI) GO TO 9
8     CONTINUE
      GO TO 11
9     PERI=2.*(TZERO(ITZ)-TZERO(ICOUNT))/FLOAT(ITZ-ICOUNT)
      OMEGA=TP/PERI
      TZERX=TZERO(ITZ)
      TEND=FLOAT(NT-1)
      AMX=A(NT)/SIN(OMEGA*(TEND-TZERX))
      PRINT 21,  ALFA,OMEGA
C
C     EXTRAPOLATE WITH ATTENUATED SINE WAVE
C
      DO 10 I=NTP,N
      TIM=FLOAT(I-1)
10    A(I)=AMX*EXP(-ALFA*(TIM-TEND))*SIN(OMEGA*(TIM-TZERX))
      NTX=N
      GO TO 14
11    PRINT 22
12    NTX=NT
C
C     NO EXTRAPOLATION
C
      DO 13 I=NTP,N
13    A(I)=0.
14    IF (NOC.NE.0) GO TO 16
C
C     APPLY HALF COSINE BELL TAPER TO LAST 1/10 OF POINTS
C
      NTP=NTX/10
      IF (NTP.LT.1) GO TO 16
      NTM=NTX-NTP+1
      TRA=PI/FLOAT(NTP+1)
      TRB=0.
      DO 15 I=NTM,NTX
      TRB=TRB+TRA
15    A(I)=A(I)*(1.+COS(TRB))*.5
16    NTP=N+1
      DO 17 I=1,N
      ITZ=NTP-I
      ITM=2*ITZ
      A(ITM)=0.
```

```
17    A(ITM-1)=A(ITZ)
      IF (M.LE.MMAX) GO TO 18
      MMAX=M
      S(1)=0.
      IFS=-1
      GO TO 19
18    IFS=-2
19    CALL FORT (A,M,S,IFS,IFERR)
      IF (IFERR.NE.0) PRINT 23,  IFERR
      IF (IFERR.EQ.1) STOP
      NTNR=N
      IF (N.GT.NTNX) NTNR=NTNX
      DO 20 I=1,NTNR
20    TRAN(I)=A(I)
      RETURN
C
C
21    FORMAT  ( 1X,84HINPUT DATA EXTRAPOLATED USING ATTENUATED SINE FUN
     1CTION EXP(-A*T)*SIN(W*T) WITH A*DT=,E12.5,10H AND W*DT=,E12.5)
22    FORMAT  (1X,27HINPUT DATA NOT EXTRAPOLATED)
23    FORMAT  (//,1X,46HERROR FLAG RETURNED BY SUBROUTINE FORT, IFERR=,I
     14,/)
24    FORMAT  ( /,1X,21HFOURIER TRANSFORM - -,/,1X,23HNUMBER OF POINTS
     1INPUT=,I7,/,1X,29HNUMBER OF POINTS TRANSFORMED=,I7)
      END
```

```
      SUBROUTINE LFACTR (A,N,IX1,IX2,P)
C
C     S/R WHICH PERFORMS GAUSS-DOOLITTLE MANIPULATIONS ON THE TWO BLOCKS
C     UNFACTORED OR PARTIALLY-FACTORED MATRIX WHICH IS IN THE CORE STORA
C
C     THE MANIPULATIONS ARE BASED ON THE GAUSS-DOOLITTLE ALGORITHM PRESE
C     PAGES 411-416 OF A. RALSTON -- A FIRST COURSE IN NUMERICAL ANALYSI
C     COMMENTS BELOW REFER TO COMMENTS IN RALSTONS TEXT.
      COMMON /MATPAR/ NBLOKS,NPBLK,NLAST,INT
      COMMON /SCHATM/ D(200)
      DIMENSION A(N,N), P(N)
      INTEGER  R,R1,R2,P,PJ,PR

      IFLG=0
C
C     INITIALIZE R1,R2,J1,J2
C
      L1=IX1.EQ.1.AND.IX2.EQ.2
      L2=(IX2-1).EQ.IX1
      L3=IX2.EQ.NBLOKS
      IF (L1) GO TO 1
      GO TO 2
1     R1=1
      R2=2*NPBLK
      J1=1
      J2=-1
      GO TO 5
2     R1=NPBLK+1
      R2=2*NPBLK
      J1=(IX1-1)*NPBLK+1
      IF (L2) GO TO 3
      GO TO 4
3     J2=J1+NPBLK-2
      GO TO 5
4     J2=J1+NPBLK-1
5     IF (L3) R2=NPBLK+NLAST
      J2P2S=J2+2
      DO 16 R=R1,R2
C
C     STEP 1
C
      DO 6 K=J1,N
      D(K)=A(K,R)
6     CONTINUE
C
C     STEPS 2 AND 3
C
      IF (L1.OR.L2) J2=J2+1
      IF (J1.GT.J2) GO TO 9
      IXJ=0
      DO 8 J=J1,J2
      IXJ=IXJ+1
      PJ=P(J)
      A(J,R)=D(PJ)
      D(PJ)=D(J)
      JP1=J+1
```

```
      DO 7 I=JP1,N
      D(I)=D(I)-A(I,IXJ)*A(J,R)
7     CONTINUE
8     CONTINUE
9     CONTINUE
C
C     STEP 4
C
      J2P1=J2+1
      IF (L1.OR.L2) GO TO 11
      IF (N.LT.J2P1) GO TO 16
      DO 10 I=J2P1,N
      A(I,R)=D(I)
10    CONTINUE
      GO TO 16
11    DMAX=ABS(D(J2P1))
      P(J2P1)=J2P1
      J2P2=J2+2
      IF (J2P2.GT.N) GO TO 13
      DO 12 I=J2P2,N
      ELMAG=ABS(D(I))
      IF (ELMAG.LT.DMAX) GO TO 12
      DMAX=ELMAG
      P(J2P1)=I
12    CONTINUE
13    CONTINUE
      IF (DMAX.LT.1.E-10) IFLG=1
      PR=P(J2P1)
      A(J2P1,R)=D(PR)
      D(PR)=D(J2P1)
C
C     STEP 5
C
      IF (J2P2.GT.N) GO TO 15
      DO 14 I=J2P2,N
      A(I,R)=D(I)/A(J2P1,R)
14    CONTINUE
15    CONTINUE
      IF (IFLG.EQ.0) GO TO 16
      PRINT 17,  J2,DMAX
      IFLG=0
16    CONTINUE
      RETURN
C
17    FORMAT (1H ,'PIVOT(',I3,')=',E16.8)
      END
```

```
      SUBROUTINE LTSOLV (A,AR,N,IX,B)
C
C     S/R TO SOLVE THE MATRIX EQUATION Y(R)*LU(T)=B(R) WHERE (R) DENOTES
C     ROW VECTOR AND LU(T) DENOTES THE LU DECOMPOSITION OF THE TRANSPOSE
C     OF THE ORIGINAL COEFFICIENT MATRIX.  THE LU(T) DECOMPOSITION IS
C     STORED ON FILE 13 IN BLOCKS IN ASCENDING ORDER AND ON FILE 14 IN
C     BLOCKS OF DESCENDING ORDER.
C
C
      COMMON /MATPAR/ NBLOKS,NPBLK,NLAST,INT
      COMMON /SCRATM/ Y(200)
      DIMENSION A(N,N), B(N), IX(N), AR(1)
C
C     FORWARD SUBSTITUTION
C
      I1=1
      I2=NPBLK*N
      J=0
      DO 4 IXBLK1=1,NBLOKS
      READ (13) (AR(I),I=I1,I2)
      K2=NPBLK
      IF (IXBLK1.EQ.NBLOKS) K2=NLAST
      DO 3 K=1,K2
      JM1=J
      J=J+1
      SUM=0.
      IF (JM1.LT.1) GO TO 2
      DO 1 I=1,JM1
      SUM=SUM+A(I,K)*Y(I)
1     CONTINUE
2     CONTINUE
      Y(J)=(B(J)-SUM)/A(J,K)
3     CONTINUE
4     CONTINUE
C
C     BACKWARD SUBSTITUTION
C
      J=N+1
      DO 8 IXBLK1=1,NBLOKS
      READ (14) (AR(I),I=I1,I2)
      K2=NPBLK
      IF (IXBLK1.EQ.1) K2=NLAST
      DO 7 K=1,K2
      KP=K2-K+1
      JP1=J
      J=J-1
      SUM=0.
      IF (N.LT.JP1) GO TO 6
      DO 5 I=JP1,N
      SUM=SUM+A(I,KP)*B(I)
5     CONTINUE
6     CONTINUE
      B(J)=Y(J)-SUM
7     CONTINUE
8     CONTINUE
C
```

```
C     UNSCRAMBLE SOLUTION
C
      DO 9 I=1,N
      IXI=IX(I)
      Y(IXI)=B(I)
9     CONTINUE
      DO 10 I=1,N
10    B(I)=Y(I)
      REWIND 13
      REWIND 14
      RETURN
      END
```

```
      SUBROUTINE LUNSCR (A,AR,N,IX,IP)
C
C     S/R WHICH UNSCRAMBLES, SCRAMBLED FACTORED MATRIX
C
      COMMON /MATPAR/ NBLOKS,NPBLK,NLAST,INT
      DIMENSION A(N,N), IP(N), IX(N), AR(1)
      I1=1
      I2=NPBLK*N
      NM1=N-1
      DO 4 IXBLK1=1,NBLOKS
      READ (11) (AR(I),I=I1,I2)
      K1=(IXBLK1-1)*NPBLK+2
      IF (NM1.LT.K1) GO TO 3
      J2=0
      DO 2 K=K1,NM1
      IF (J2.LT.NPBLK) J2=J2+1
      IPK=IP(K)
      DO 1 J=1,J2
      TEMP=A(K,J)
      A(K,J)=A(IPK,J)
      A(IPK,J)=TEMP
1     CONTINUE
2     CONTINUE
3     CONTINUE
      WRITE (13) (AR(I),I=I1,I2)
4     CONTINUE
      DO 5 IXBLK1=1,NBLOKS
      BACKSPACE  13
      IF (IXBLK1.NE.1) BACKSPACE  13
      READ (13) (AR(I),I=I1,I2)
      WRITE (14) (AR(I),I=I1,I2)
5     CONTINUE
      DO 6 I=1,N
      IX(I)=I
6     CONTINUE
      DO 7 I=1,N
      IPI=IP(I)
      IXT=IX(I)
      IX(I)=IX(IPI)
      IX(IPI)=IXT
7     CONTINUE
      REWIND  11
      REWIND  13
      REWIND  14
      RETURN
      END
```

```
      SUBROUTINE MOVE (ROX,ROY,ROZ,XS,YS,ZS,ITS,NRPT,ITGI)
C
C     SUBROUTINE MOVE MOVES THE STRUCTURE WITH RESPECT TO ITS
C     COORDINATE SYSTEM OR REPRODUCES STRUCTURE IN NEW POSITIONS.
C     STRUCTURE IS ROTATED ABOUT X,Y,Z AXES BY ROX,ROY,ROZ
C     RESPECTIVELY, THEN SHIFTED BY XS,YS,ZS
C
      COMMON /DATA/ N,NP,X(200),Y(200),Z(200),SI(200),BI(200),ALP(200),
     1 BET(200),ICON1(200),ICON2(200),ITAG(200),IPX,IPY,IPZ
      DIMENSION X2(1), Y2(1), Z2(1)
      EQUIVALENCE (X2(1),SI(1)), (Y2(1),ALP(1)), (Z2(1),BET(1))
      SPS=SIN(ROX)
      CPS=COS(ROX)
      STH=SIN(ROY)
      CTH=COS(ROY)
      SPH=SIN(ROZ)
      CPH=COS(ROZ)
      XX=CPH*CTH
      XY=CPH*STH*SPS-SPH*CPS
      XZ=CPH*STH*CPS+SPH*SPS
      YX=SPH*CTH
      YY=SPH*STH*SPS+CPH*CPS
      YZ=SPH*STH*CPS-CPH*SPS
      ZX=-STH
      ZY=CTH*SPS
      ZZ=CTH*CPS
      I1=ISEGNO(ITS,1)
      IX=I1
      NRP=NRPT
      IF (NRPT.EQ.0) NRP=1
      K=N
      IF (NRPT.EQ.0) K=I1-1
      DO 2 IR=1,NRP
      DO 1 I=I1,N
      K=K+1
      XI=X(I)
      YI=Y(I)
      ZI=Z(I)
      X(K)=XI*XX+YI*XY+ZI*XZ+XS
      Y(K)=XI*YX+YI*YY+ZI*YZ+YS
      Z(K)=XI*ZX+YI*ZY+ZI*ZZ+ZS
      XI=X2(I)
      YI=Y2(I)
      ZI=Z2(I)
      X2(K)=XI*XX+YI*XY+ZI*XZ+XS
      Y2(K)=XI*YX+YI*YY+ZI*YZ+YS
      Z2(K)=XI*ZX+YI*ZY+ZI*ZZ+ZS
      BI(K)=BI(I)
      ITAG(K)=ITAG(I)+ITGI
1     CONTINUE
      I1=N+1
      N=K
2     CONTINUE
      RETURN
      END
```

```
      SUBROUTINE RFLD (THET,PHI,ETH,EPH,TSTRT,KLM)
C
C     RFLD COMPUTES THE RADIATED FIELD IN THE DIRECTION THETA, PHI FOR
C     ALL TIME STEPS POSSIBLE WITH THE AVAILABLE CURRENTS
C
      COMMON /DATA/ N,NP,X(200),Y(200),Z(200),SI(200),BI(200),ALP(200),
     1   BET(200),ICON1(200),ICON2(200),ITAG(200),IPX,IPY,IPZ
      COMMON /SCOMP/ SX(200),SY(200),SZ(200)
      COMMON /INTERP/ AT(3,200),BT(3,200),CT(3,200),ES(3),FS(3),GS(3),
     1   E(3),H(3)
      COMMON /CONST/ CDT,VEL,DT,NTSTEP
      COMMON /ARRAX/ QI1,QI2,QI3,ET(200,5,8),EP(200,5,8),IRET(200,8)
      COMMON /ARRAY/ CO(6400)
      COMMON /IOFLG/ NCFMX,NCQMX,IOC,IOCQ,NTMAX,NBOUT,JP1,JP2
      DIMENSION A(3), B(3), C(3), P(3), ETH(1), EPH(1), ALFA(3,3)
      N2=N*2
      STH=SIN(THET)
      CTH=COS(THET)
      SP=SIN(PHI)
      CP=COS(PHI)
      ERX=STH*CP
      ERY=STH*SP
      ERZ=CTH
      THX=CTH*CP
      THY=CTH*SP
      THZ=-STH
      PHX=-SP
      PHY=CP
      KMN=9999
      IPLAN=0
      LPX=1
      LPY=1
      LPZ=1
      IF (IPX.NE.0) LPX=2
      IF (IPY.NE.0) LPY=2
      IF (IPZ.NE.0) LPZ=2
C
C     BEGIN LOOP OVER SEGMENTS, INCLUDING IMAGES, TO COMPUTE
C     COEFFICIENTS FOR RADIATED FIELD
C
      DO 17 JX=1,LPX
      RFX=FLOAT(3-JX*2)
      SFX=RFX
      IF (IPX.LT.0) SFX=1.
      DO 17 JY=1,LPY
      RFY=FLOAT(3-JY*2)
      SFY=RFY
      IF (IPY.LT.0) SFY=1.
      DO 17 JZ=1,LPZ
      RFZ=FLOAT(3-JZ*2)
      SFZ=RFZ
      IF (IPZ.LT.0) SFZ=1.
      RFL=SFX*SFY*SFZ
      RFL=-1.E-7*RFL
      IPLAN=IPLAN+1
      DO 1 J=1,N
```

```
      AL=-(ERX*X(J)*RFX+ERY*Y(J)*RFY+ERZ*Z(J)*RFZ)
      TRET=AL/VEL
      K=TRET/DT+SIGN(.5,TRET)
      IRET(J,IPLAN)=K
      IF (K.LT.KMN) KMN=K
      DO 1 M=1,5
      ET(J,M,IPLAN)=0.
1     EP(J,M,IPLAN)=0.
      DO 16 J=1,N
      XJ=X(J)*RFX
      YJ=Y(J)*RFY
      ZJ=Z(J)*RFZ
      SXJ=SX(J)*RFX
      SYJ=SY(J)*RFY
      SZJ=SZ(J)*RFZ
      AL=-(ERX*XJ+ERY*YJ+ERZ*ZJ)
      BA=-(ERX*SXJ+ERY*SYJ+ERZ*SZJ)
      K=IRET(J,IPLAN)
      TAU=DT*FLOAT(K)
      DO 2 L=1,3
      A(L)=AT(L,J)
      B(L)=BT(L,J)
      C(L)=CT(L,J)
2     P(L)=2.*ES(L)*TAU+FS(L)
      EL=SI(J)
      ELC=EL*EL*EL/12.
      SDOTT=SXJ*THX+SYJ*THY+SZJ*THZ
      SDOTP=SXJ*PHX+SYJ*PHY
      DO 3 M=1,3
      T1=H(M)*AL+P(M)
      T2=H(M)*BA
      DO 3 L=1,3
3     ALFA(L,M)=((A(L)*T1+B(L)*T2)*ELC+C(L)*T1*EL)*RFL
      JC1=ICON1(J)
      IF (JC1.NE.0) GO TO 4
      KKM=0
      GO TO 8
4     IF (JC1.LT.19000) GO TO 5
      SIG1=FLOAT(JC1-20000)
      JC1=J
      GO TO 7
5     IF (ICON2(JC1).NE.J) GO TO 6
      SIG1=1.
      GO TO 7
6     IF (ICON1(JC1).NE.J) GO TO 23
      SIG1=-1.
7     KM=IRET(JC1,IPLAN)
      KKM=KM-K
      IF (IABS(KKM).LE.1) GO TO 8
      PRINT 24, J,JC1,KKM
      STOP
8     JC2=ICON2(J)
      IF (JC2.NE.0) GO TO 9
      KKP=0
      GO TO 13
9     IF (JC2.LT.19000) GO TO 10
      SIG2=FLOAT(JC2-20000)
      JC2=J
      GO TO 12
10    IF (ICON1(JC2).NE.J) GO TO 11
```

```
      SIG2=1.
      GO TO 12
11    IF (ICON2(JC2).NE.J) GO TO 23
      SIG2=-1.
12    KP=IRET(JC2,IPLAN)
      KKP=KP-K
      IF (IABS(KKP).LE.1) GO TO 13
      PRINT 24, J,JC2,KKP
      STOP
13    DO 15 M=1,5
      KXX=M+1
      IF (JC1.EQ.0) GO TO 14
      KMX=KKM+KXX
      ET(JC1,KMX,IPLAN)=ET(JC1,KMX,IPLAN)+ALFA(1,M)*SDOTT*SIG1
      EP(JC1,KMX,IPLAN)=EP(JC1,KMX,IPLAN)+ALFA(1,M)*SDOTP*SIG1
14    ET(J,KXX,IPLAN)=ET(J,KXX,IPLAN)+ALFA(2,M)*SDOTT
      EP(J,KXX,IPLAN)=EP(J,KXX,IPLAN)+ALFA(2,M)*SDOTP
      IF (JC2.EQ.0) GO TO 15
      KPX=KKP+KXX
      ET(JC2,KPX,IPLAN)=ET(JC2,KPX,IPLAN)+ALFA(3,M)*SDOTT*SIG2
      EP(JC2,KPX,IPLAN)=EP(JC2,KPX,IPLAN)+ALFA(3,M)*SDOTP*SIG2
15    CONTINUE
16    CONTINUE
17    CONTINUE
C
C     COMPUTE RADIATED FIELD AT ALL TIME STEPS POSSIBLE WITH SUMMATION
C     PROCEEDING IN ORDER OF RETARDED TIMES
C
      KMN=KMN-1
      KKM=1
      KKP=NTSTEP
      KM=KKM+KMN
      KP=KKP+KMN
      KXX=KM-1
      KLM=KP-KXX
      TSTRT=DT*FLOAT(KXX)
      DO 18 K=1,KLM
      ETH(K)=0.
18    EPH(K)=0.
      KTS=1
      NOCQ=9999999
      IF (IOCQ.EQ.0) GO TO 19
      REWIND 11
      READ (11) NOCQ,(CQ(J),J=1,NOCQ)
19    DO 22 KC=KKM,KKP
20    KCB=(KC-1)*N2-KTS
      IF ((KCB+1).LT.NOCQ) GO TO 21
      KTS=KTS+NOCQ
      READ (11) NOCQ,(CQ(J),J=1,NOCQ)
      GO TO 20
21    KKB=KC-KXX+3
      DO 22 IPLNE=1,IPLAN
      DO 22 J=1,N
      KCC=KCB+2*J
      K=KKB+IRET(J,IPLNE)
      DO 22 M=1,5
      KPX=K-M
      IF (KPX.LT.1) GO TO 22
      IF (KPX.GT.KLM) GO TO 22
      ETH(KPX)=ETH(KPX)+ET(J,M,IPLNE)*CQ(KCC)
```

```
      EPH(KPX)=EPH(KPX)+EP(J,M,IPLNE)*CQ(KCC)
22    CONTINUE
      RETURN
23    PRINT 25
      STOP
C
24    FORMAT (  /,1X,26HRETARDED TIMES TO SEGMENTS,I5,4H AND,I5,
     1  10H DIFFER BY,I5,11H TIME STEPS)
25    FORMAT (/,1X,62H***ERROR - ROUTINE RFLD-- INCONSISTENT SEGMENT CON
     1NECTION DATA)
      END
```

```
      SUBROUTINE RFPAT (ITRAN,IPCH,NTH,NPH,THET,PHI,DTH,DPH)
C
C     RFPAT CALLS RFLD TO COMPUTE THE RADIATED FIELD AT A SET OF ANGLES,
C     THETA AND PHI, AND PRINTS THE RESULTS.  THE COMPUTED FIELD CAN
C     ALSO BE FOURIER TRANSFORMED TO THE FREQUENCY DOMAIN
C
      COMMON /ESORC/  ESORC(1024),IFST,NTRAN,MTRAN,NTNX,DFRQ,ENIN,ENRD,I
     1FEN
      COMMON /CMAT/  CURF(1600),IP(200),IX(200)
      COMMON /CONST/  CDT,VEL,DT,NTSTEP
      COMMON /SCRATM/ PWR(512)
      DIMENSION ETH(1024), EPH(1024), ETC(512), EPC(512), ESC(512), IPOL
     1(3)
      COMPLEX   TPJ,CARG,FAZ,ETHC,EPHC,ETC,EPC,ESC
      EQUIVALENCE (ETH,CURF), (ETC,CURF), (EPC,EPH), (ESC,ESORC)
      DATA    TA,TSMIN,TPJ/,017453292,5.,(0.,6.2831853)/
      DATA    IPOL/6HLINEAR,5HRIGHT,4HLEFT/,IBLK/1H /
      TSTRT=0.
      NFLD=0
      TH=THET-DTH
      DO 15 IT=1,NTH
      TH=TH+DTH
      THA=TH*TA
      PH=PHI-DPH
      DO 15 IPH=1,NPH
      PH=PH+DPH
      PHA=PH*TA
      PRINT 18,  TH,PH
      CALL RFLD (THA,PHA,ETH,EPH,TSTRT,NFLD)
      NPRT=NFLD/2
      NADD=NPRT
      IF (2*NPRT.NE.NFLD) NADD=NPRT+1
      TIM=TSTRT-DT
      TADD=DT*FLOAT(NADD)
      DO 1 I1=1,NPRT
      I2=I1+NADD
      TIM=TIM+DT
      TIM2=TIM+TADD
      POL1=ATGN2(EPH(I1),ETH(I1))/TA
      POL2=ATGN2(EPH(I2),ETH(I2))/TA
1     PRINT 19,  I1,TIM,ETH(I1),EPH(I1),POL1,I2,TIM2,ETH(I2),EPH(I2),POL
     12
      IF (NADD.EQ.NPRT) GO TO 2
      POL1=ATGN2(EPH(NADD),ETH(NADD))/TA
      TIM=TSTRT+DT*FLOAT(NPRT)
      PRINT 19,  NADD,TIM,ETH(NADD),EPH(NADD),POL1
C
C     COMPUTE RADIATED ENERGY
C
2     F=0.
      DO 3 I=1,NFLD
3     F=F+ETH(I)*ETH(I)+EPH(I)*EPH(I)
      F=F*DT/376.73
      PRINT 21,  F,TH,PH
      IF (IFEN.EQ.0) GO TO 4
      POL1=F/ENIN
```

```
      POL2=F/ENRD
      IF (POL1.LT.1.E-20) GO TO 4
      POL1=10.*ALOG10(POL1)
      POL2=10.*ALOG10(POL2)
      PRINT 20,  POL1,POL2
4     IF (IPCH.EQ.0) GO TO 5
C
C     PUNCH FIELD VALUES
C
      PUNCH 26, TSTRT,DT
      PUNCH 22,  NFLD,TH,PH
      PUNCH 23,  (ETH(I),I=1,NFLD)
      PUNCH 24,  NFLD,TH,PH
      PUNCH 23,  (EPH(I),I=1,NFLD)
5     IF (ITRAN.EQ.0) GO TO 15
C
C     FOURIER TRANSFORM TO OBTAIN FREQUENCY RESPONSE
C
      IF (IFST.EQ.1) GO TO 7
      MTRAN=ALOG(FLOAT(NTRAN))/.69314718+1.5
      CALL ITOF (ESORC,NTRAN,MTRAN,1,1,NTNX,NTNR)
      IFST=1
      DFRQ=1./(DT*FLOAT(2**MTRAN))
      NTNR=NTNR/2
      DO 6 I=1,NTNR
6     PWR(I)=0.
7     CALL ITOF (ETH,NTRAN,MTRAN,0,0,NTNX,NTNR)
      CALL ITOF (EPH,NTRAN,MTRAN,0,0,NTNX,NTNR)
      NTNR=NTNR/2
      FMAX=1./(TSMIN*DT)
      CARG=-TPJ*TSTRT
      PRINT 17
      IF (ITRAN.GT.1) PUNCH 25, TH,PH
      F=-DFRQ
      DO 14 I=1,NTNR
      F=F+DFRQ
      IF (F.GT.FMAX) GO TO 15
      FQ=F*1.E-6
      FAZ=CEXP(CARG*F)/ESC(I)
      ETHC=ETC(I)*FAZ
      EPHC=EPC(I)*FAZ
      ETHM2=REAL(ETHC*CONJG(ETHC))
      EPHM2=REAL(EPHC*CONJG(EPHC))
      ETHM=SQRT(ETHM2)
      EPHM=SQRT(EPHM2)
      ETHA=ATGN2(AIMAG(ETHC),REAL(ETHC))/TA
      EPHA=ATGN2(AIMAG(EPHC),REAL(EPHC))/TA
      IF (ETHM2.GT.1.E-20.OR.EPHM2.GT.1.E-20) GO TO 8
      TILTA=0.
      EMAJR2=0.
      EMINR2=0.
      AXRAT=0.
      ISENS=IBLK
      GO TO 13
8     DFAZ=EPHA-ETHA
      IF (EPHA.LT.0.) GO TO 9
      DFAZ2=DFAZ-360.
      GO TO 10
9     DFAZ2=DFAZ+360.
10    IF (ABS(DFAZ).GT.ABS(DFAZ2)) DFAZ=DFAZ2
```

```
      CDFAZ=COS(DFAZ*TA)
      TS1=ETHM2-EPHM2
      TS2=2.*EPHM*ETHM*CDFAZ
      TILTA=.5*ATAN2(TS2,TS1)
      STILTA=SIN(TILTA)
      TS1=TS1*STILTA*STILTA
      TS2=TS2*STILTA*COS(TILTA)
      EMAJR2=-TS1+TS2+ETHM2
      EMINR2=TS1-TS2+EPHM2
      IF (EMINR2.LT.0.) EMINR2=0.
      AXRAT=SQRT(EMINR2/EMAJR2)
      TILTA=TILTA/TA
      IF (AXRAT.GT.1.E-5) GO TO 11
      ISENS=IPOL(1)
      GO TO 13
11    IF (DFAZ.GT.0.) GO TO 12
      ISENS=IPOL(2)
      GO TO 13
12    ISENS=IPOL(3)
13    WLAM=F
      IF (WLAM.LT.1.E-25) WLAM=1.E-25
      WLAM=2.998E+8/WLAM
      POR=PWR(I)
      IF (POR.LT.1.E-25) POR=.0013273*WLAM*WLAM
      GTOT=.01668*(ETHM2+EPHM2)/POR
      IF (GTOT.LT.1.E-20) GTOT=1.E-20
      GTOT=10.*ALOG10(GTOT)
      ETHM=ETHM/WLAM
      EPHM=EPHM/WLAM
      PRINT 16,  I,FQ,ETHM,ETHA,EPHM,EPHA,AXRAT,TILTA,ISENS,GTOT
      IF (ITRAN.GT.1) PUNCH 16, I,FQ,ETHM,ETHA,EPHM,EPHA
14    CONTINUE
15    CONTINUE
      RETURN
C
C
16    FORMAT  (1X,I5,E12.4,E13.4,F9.3,E15.4,F9.3,F11.5,F9.2,2X,A6,F14.3)
17    FORMAT  (  ///,33X,43H- - - FREQUENCY DOMAIN RADIATED FIELD - - -,
     1  //,2X,4HSTEP,2X,9HFREQUENCY,3X,20H- - - E(THETA) - - -,4X,
     2  18H- - - E(PHI) - - -,6X,24H- - - POLARIZATION - - -,5X,
     3         10HPOWER GAIN,/,3X,3HNO.,4X,5H(MHZ),6X,9HMAGNITUDE,4X,
     4  5HPHASE,6X,9HMAGNITUDE,4X,5HPHASE,7X,5HAXIAL,5X,4HTILT,3X,
     5  5HSENSE,9X,4H(DB),/,22X,7HVOLTS/M,4X,7HDEGREES,6X,7HVOLTS/M,
     6  4X,7HDEGREES,6X,5HRATIO,5X,4HDEG.)
18    FORMAT  (  //,42X,30H- - - - RADIATED FIELD - - - -,//,47X,6HTHETA
     1=, F8.3,/,47X,6HPHI  =,F8.3,//,2(2X,4HSTEP,6X,4HTIME,8X,20HELECTRI
     2C FIELD (V/M),4X,4HPOL.,8X),/,2(3X,3HNO.,5X,6H(SEC.),8X,5HTHETA,9X
     3,  3HPHI,5X,6H(DEG.),7X))
19    FORMAT  (2(1X,I5,E13.5,E13.4,E12.4,F9.3,7X))
20    FORMAT  (  1X,24HTIME DOMAIN POWER GAIN =,F7.2,3H DB,/,1X,28HTIME
     1DOMAIN DIRECTIVE GAIN =,F7.2,3H DB,//)
21    FORMAT  (  //,1X,23HTOTAL ENERGY RADIATED =,E11.4,17H JOULES/STERA
     1DIAN,3X,7H(THETA=,F8.3,12H DEG.,  PHI=,F8.3,6H DEG.))
22    FORMAT  (I5,2X,26HRADIATED FIELDS FOR THETA=,F8.3,11H DEG., PHI=,F
     18.3,18H DEG.,  THETA POL.)
23    FORMAT  (6E12.5)
24    FORMAT  (I5,2X,26HRADIATED FIELDS FOR THETA=,F8.3,11H DEG., PHI=,F
     18.3,16H DEG.,  PHI POL.)
25    FORMAT (23HRADIATED FIELD.  THETA=,F10.5,6H  PHI=,F10.5)
26    FORMAT (14HSTARTING TIME=,E12.5,3X,15HTIME INCREMENT=,E12.5)
      END
```

```
      SUBROUTINE SECOND (T)
C
C     RETURNS CURRENT RUNNING TIME IN SECONDS
C
      CALL TUSEOM (I)
      T=I/1000.
      RETURN
      END
```

```
      SUBROUTINE SOLVE (N,A,P,B,NDIM)
C
C     SUBROUTINE TO SOLVE THE MATRIX EQUATION LU*X=B WHERE L IS A UNIT
C     LOWER TRIANGULAR MATRIX AND U IS AN UPPER TRIANGULAR MATRIX BOTH
C     OF WHICH ARE STORED IN A.  THE RHS VECTOR B IS INPUT AND THE
C     SOLUTION IS RETURNED THROUGH VECTOR B.      (MATRIX TRANSPOSED)
C
      DIMENSION A(NDIM,NDIM), P(NDIM), B(NDIM)
      COMMON /SCRATM/ Y(200)
      INTEGER  P,PI
C
C     FORWARD SUBSTITUTION
C
      DO 3 I=1,N
      PI=P(I)
      Y(I)=B(PI)
      B(PI)=B(I)
      IP1=I+1
      IF (IP1.GT.N) GO TO 2
      DO 1 J=IP1,N
      B(J)=B(J)-A(I,J)*Y(I)
1     CONTINUE
2     CONTINUE
3     CONTINUE
C
C     BACKWARD SUBSTITUTION
C
      DO 6 K=1,N
      I=N-K+1
      SUM=0.
      IP1=I+1
      IF (IP1.GT.N) GO TO 5
      DO 4 J=IP1,N
      SUM=SUM+A(J,I)*B(J)
4     CONTINUE
5     CONTINUE
      B(I)=(Y(I)-SUM)/A(I,I)
6     CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TSOL (CURF,IX,IP,N)
C
C     TSOL COMPUTES STRUCTURE CURRENTS
C
      COMMON /CONST/  CDT,VEL,DT,NTSTEP
      COMMON /EINC/  EINC(200),ESRC
      COMMON /ARRAY/  CQ(6400)
      COMMON /ARRAX/  QI1,QI2,QI3,EC(200,5),EQ(200,5),IRET(200),BFR(1540
     10)
      COMMON /IOFLG/  NCFMX,NCQMX,IOC,IOCQ,NTMAX,NBOUT,JP1,JP2
      COMMON /MATPAR/  NBLOKS,NPBLK,NLAST,INT
      COMMON /ESORC/  ESORC(1024),IFST,NTRAN,MTRAN,NTNX,DFRQ,ENIN,ENRD,I
     1FEN
      DIMENSION ICOF(17600), COF(17600)
      EQUIVALENCE (COF,EC), (ICOF,EC)
      N2=N*2
      N4P=2*N2+1
      IC1=N2
      IC2=N4P
      ICL=9999999
      IOLIM=9999999
      LCQ=NCQMX/N2
      NOCQ=LCQ-NTMAX+1
      IF (NOCQ.LT.1) GO TO 1
      LCQ=LCQ*N2
      NOCQ=NOCQ*N2
      L1=1
      L2=NOCQ
      NSFT=LCQ-NOCQ
      PRINT 36
      GO TO 2
    1 NOCQ=(LCQ/2)*N2
      IF (NOCQ.LT.N2) GO TO 35
      LCQ=2*NOCQ
      L1=NOCQ+1
      L2=LCQ
      PRINT 37
    2 IOCQ=0
      NBAS=0
      ICFLG=0
      NELO=0
      NBOUT=0
      PRINT 38,  NOCQ,NTMAX
      PRINT 39
      PRINT 40
      DO 3 I=2,N2,2
    3 CQ(I)=0.
      TIME=-DT
C
C     BEGIN TIME LOOP
C
      DO 31 ITIME=1,NTSTEP
      TIME=TIME+DT
      ITLOC=(ITIME-1)*N2
C
C     SET UP APPLIED FIELD AT PRESENT TIME
```

```
C
      CALL EINCF (TIME)
      IF (ITIME.EQ.1) GO TO 18
      IF (ICFLG.EQ.0) GO TO 5
C
C     COMPUTE FIELD AT PRESENT TIME DUE TO CURRENTS AT RETARDED TIMES
C
      NBKS=((NTMAX-K)*N2+NELO+NOCQ-1)/NOCQ+1
      IF (NBKS.GT.NBOUT) NBKS=NBOUT
      NBAS=NELO-NBKS*NOCQ+NOCQ
      DO 4 J=1,NBKS
4     BACKSPACE  11
      READ (11)  NIN,(CQ(J),J=1,NOCQ)
      NBIN=1
5     IF (IOC.EQ.0) GO TO 7
      REWIND  12
6     READ (12)  ICL,LEND,(COF(J),J=1,ICL)
      IF (ITIME.LE.LEND) GO TO 6
7     ESCAT=0.
      ICF=1
      I=1
8     INDX=ICOF(ICF)
      ICF=ICF+1
      IF (INDX.LT.0) GO TO 9
      ICF=ICF+INDX/10000
      GO TO 8
9     IF (ITIME.LE.(-INDX/10000)) GO TO 8
      GO TO 11
10    EINC(I)=EINC(I)+ESCAT
      ESCAT=0.
11    IF (INDX.EQ.0) GO TO 18
      INDX=-INDX
      K=INDX/10000
      I=INDX-K*10000
      K=ITIME-K
12    ICQB=(K-1)*N2-NBAS
      IF (ICQB.LT.IOLIM) GO TO 14
      IF (NBIN.EQ.NBKS) GO TO 14
      READ (11)  NIN,(CQ(J),J=1,NOCQ)
      NBIN=NBIN+1
      NBAS=NBAS+NOCQ
      GO TO 12
13    IF (ICF.LE.ICL) GO TO 14
      READ (12)  ICL,LEND,(COF(J),J=1,ICL)
      ICF=1
14    INDX=ICOF(ICF)
      ICF=ICF+1
      IF (INDX.LE.0) GO TO 10
      J1=INDX/10000
      J2=INDX-J1*10000
      ICQ=ICQB+J2
      IF (K.EQ.ITIME) GO TO 16
      DO 15 J=1,J1
      ESCAT=ESCAT+COF(ICF)*CQ(ICQ)
      ICQ=ICQ+1
15    ICF=ICF+1
      GO TO 13
16    ICQ=ICQ+1
      ICF=ICF+1
      DO 17 J=1,J1,2
```

```
      ESCAT=ESCAT+COF(ICF)*CQ(ICQ)
      ICQ=ICQ+2
17    ICF=ICF+2
      ICF=ICF-1
      GO TO 13
C
C     SOLVE MATRIX EQUATION FOR CURRENTS AT PRESENT TIME
C
18    IF (INT.EQ.0) GO TO 19
      CALL LTSOLV (CURF,CURF,N,IX,EINC)
      GO TO 20
19    CALL SOLVE (N,CURF,IP,EINC,N)
20    PRINT 41,  ITIME,TIME,(EINC(I),I=JP1,JP2)
      PRINT 43, ESRC
      IF (ITIME.GT.NTRAN) GO TO 21
      ESRC(ITIME)=ESRC
21    ICQ=ITLOC-1-NELO
      DO 22 I=1,N
      ICQ=ICQ+2
      ICQB=ICQ+1
      CQ(ICQ)=EINC(I)
22    CQ(ICQB)=CQ(ICQB)+QI3*EINC(I)
      IF (ITIME.EQ.NTSTEP) GO TO 31
      IF (ICQB.LT.LCQ) GO TO 26
      IF (IOCQ.EQ.1) GO TO 23
      IOCQ=1
      REWIND   11
      IF (L1.EQ.1) GO TO 23
      WRITE (11)  NOCQ,(CQ(I),I=1,NOCQ)
      NBOUT=NBOUT+1
      ICFLG=1
      IOLIM=NOCQ
23    WRITE (11)  NOCQ,(CQ(I),I=L1,L2)
      NBOUT=NBOUT+1
      NELO=NELO+NOCQ
      IF (L1.EQ.1) GO TO 24
      IC1=N2-NOCQ
      IC2=N4P-NOCQ
      GO TO 26
24    DO 25 I=1,NSFT
      INOX=I+NOCQ
25    CQ(I)=CQ(INOX)
      NBAS=NELO
26    ICQ=ITLOC-NELO+N2
C
C     COMPUTE COMPONENT OF TIME INTEGRAL OF CURRENT AT NEXT TIME STEP
C     THAT DEPENDS ON PRESENT AND PAST VALUES OF CURRENT AND
C     TIME INTEGRAL OF CURRENT
C
      IF (ITIME.EQ.1) GO TO 28
      DO 27 J=1,N
      ICQ=ICQ+2
      ICQX=ICQ-IC1
      ICQY=ICQ-IC2
27    CQ(ICQ)=CQ(ICQX)+QI1*CQ(ICQY)+QI2*EINC(J)
      GO TO 30
28    DO 29 J=1,N
      ICQ=ICQ+2
      ICQX=ICQ-IC1
29    CQ(ICQ)=CQ(ICQX)+QI2*EINC(J)
```

```
30    IF (IC1.EQ.N2) IC2=N4P
      IC1=N2
31    CONTINUE
      IF (IOCQ.EQ.0) GO TO 34
32    IF (L2.GE.ICQB) GO TO 33
      WRITE (11)  NOCQ,(CQ(I),I=L1,L2)
      NBOUT=NBOUT+1
      L1=L2+1
      L2=L2+NOCQ
      GO TO 32
33    NOCQ=ICQB-L1+1
      WRITE (11)  NOCQ,(CQ(I),I=L1,ICQB)
      NBOUT=NBOUT+1
      REWIND   11
34    IF (IOC.NE.0) REWIND   12
      RETURN
35    PRINT 42,  N
      STOP
C
C
36    FORMAT  (    1X,49HSOLUTION WILL USE CURRENTS FROM CORE STORAGE ONL
     1Y)
37    FORMAT  (    1X,44HSOLUTION WILL USE CURRENTS FROM FILE STORAGE)
38    FORMAT  (1X,33HLENGTH OF CURRENT OUTPUT BLOCK IS,I8,1X,6HWORDS,,/,
     11X,42HNUMBER OF TIME STEPS USED DURING SOLUTION=,I8,//)
39    FORMAT  (    44X,32H- - - - CURRENT SOLUTION - - - -,/)
40    FORMAT  (1X,4HSTEP,6X,4HTIME,5X,52HSEGMENT CURRENTS IN AMPS (READ
     1ACROSS) - - - - -,/,2X,3HNO.,5X,6H(SEC.),4X,52HLAST NUMBER IN
     2EACH BLOCK REPRESENTS SOURCE STRENGTH)
41    FORMAT  (/,1X,I4,E12.5,10E11.3,/,(17X,10E11.3))
42    FORMAT  (1X,72H***ERROR - ROUTINE TSOL-- INSUFFICIENT STORAGE FOR
     1SEGMENT CURRENTS.  N=,I10)
43    FORMAT (19X,10HEXCITATION,E13.4)
      END
```

```
      SUBROUTINE WIRE (XW1,YW1,ZW1,XW2,YW2,ZW2,RAD,NS,ITG)
C
C     SUBROUTINE WIRE GENERATES SEGMENT GEOMETRY DATA FOR A STRAIGHT
C     WIRE OF NS SEGMENTS.
C
      COMMON /DATA/ N,NP,X(200),Y(200),Z(200),SI(200),BI(200),ALP(200),
     1   BET(200),ICON1(200),ICON2(200),ITAG(200),IPX,IPY,IPZ
      DIMENSION X2(1), Y2(1), Z2(1)
      EQUIVALENCE (X2(1),SI(1)), (Y2(1),ALP(1)), (Z2(1),BET(1))
      IST=N+1
      N=N+NS
      NP=N
      IF (NS.LT.1) RETURN
      FNS=NS
      XD=(XW2-XW1)/FNS
      YD=(YW2-YW1)/FNS
      ZD=(ZW2-ZW1)/FNS
      XS1=XW1
      YS1=YW1
      ZS1=ZW1
      DO 1 I=IST,N
      ITAG(I)=ITG
      XS2=XS1+XD
      YS2=YS1+YD
      ZS2=ZS1+ZD
      X(I)=XS1
      Y(I)=YS1
      Z(I)=ZS1
      X2(I)=XS2
      Y2(I)=YS2
      Z2(I)=ZS2
      BI(I)=RAD
      XS1=XS2
      YS1=YS2
1     ZS1=ZS2
      RETURN
      END
```

APPENDIX B
DIAGNOSTICS

1. BLOCKING N_1 N_2 N_3

   Routine: FBLOCK

   This is printed when matrix factorization and solution requires file storage. It does not indicate an error.

   N_1 = number of blocks on file storage

   N_2 = number of matrix columns per block

   N_3 = number of matrix columns in last block ($N_3 \leq N_2$)

2. CHECK DATA, PARAMETER SPECIFYING SEGMENT POSITION IN A GROUP OF EQUAL TAGS MUST NOT BE ZERO

   Routine: ISEGNO

   The number following the tag number reference on a data card was zero. Execution terminated.

3 ERROR FLAG RETURNED BY SUBROUTINE FORT, IFERR = ---

   Routine: ITOF

   This indicates a program malfunction. The meaning of IFERR is explained in the comments in subroutine FORT.

4. ERROR - ROUTINE RFLD -- INCONSISTENT SEGMENT CONNECTION DATA

   Routine: RFLD

   This indicates a program malufnction. The connection information in arrays ICON1 and ICON2 was found to be contradictory. A possible cause is an overflow of an array.

5. ERROR - ROUTINE TSOL -- INSUFFICIENT STORAGE FOR SEGMENT CURRENTS. N = ---

   Routine: TSOL

   The dimension of array CQ must be at least 4 x N where N is the number of segments. This dimension is presently 6400 for the GE 635/645 computer and 5184 for the CDC 3800.

6. GEOMETRY DATA CARD ERROR

Routine: DATAGN

An invalid mnemonic was found on a card where a geometry data card was expected.

7. INCORRECT LABEL FOR A COMMENT CARD

Routine: MAIN

An invalid mnemonic was found on a card where a comment card was expected.

8. INPUT ENERGY NOT COMPUTED SINCE SOURCE HAS BEEN TRANSFORMED

Routine: MAIN

Request for energy budget must preceed any requests involving Fourier transforms since the Fourier transform writes over the record of the source time function.

9. INTEGRATION ATTEMPTED OVER SINGULARITY.

B, C = ---

Singularity due to R going to zero in range of integration ($R = \left[s^2 + Bs + C\right]^{1/2}$). Possible cause is overlapping segments in geometry specification.

10. INVALID DATA CARD LABEL AFTER SOLUTION

Routine: MAIN

An invalid mnemonic was found on a card where a data request card was expected. Valid mnemonics are EB, PC, AT, RF, NX, and EN

11. NO SEGMENT HAS AN ITAG OF ---

Routine: ISEGNO

TAG number used to refer to a segment does not exist.

12. NUMBER OF EXCITATION VALUES EXCEEDS ARRAY DIMENSION

Number of voltage sources cannot be greater than the dimensions of EMAG and ISRC. Dimension is now 10.

13. OVERFLOW IN FILLING ARRAY COF

Routine: COFS

This indicates a program malfunction. May possibly be corrected by increasing IST at statement CF 34A in subroutine COFS to cause more frequent dumping of data to file.

14. OVERFLOW IN FILLING ARRAY SRT

Routine COFS

Same as 13.

15. PIVOT(---) = ---

Routine: FACTR or LFACTR

This will be printed during Gauss Doolittle factoring of the interaction matrix when a pivot element less than 10^{-10} is encountered, and indicates that the matrix is nearly singular. The number in parenthesis shows on which pass through the matrix the condition occurred. This is usually an abnormal condition although execution will continue. It may result from coinciding segments or a segment of zero length.

16. RETARDED TIMES FROM SEGMENT --- TO SEGMENTS --- AND--- DIFFER BY---TIME STEPS

Routine: COFS

Retarded times from a point to two segments which are connected to each other cannot differ by more than one time step. This is usually caused by a segment with length greater than the time increment multiplied by the velocity of light.

17. RETARDED TIMES TO SEGMENTS ---AND---DIFFER BY---TIME STEPS

Routine: RFLD

Same as 16.

18. SEGMENT --- LIES IN OR BEHIND SYMMETRY PLANE

Routine: CONECT

All segments must lie on the positive side of any symmetry planes.

19. SEGMENT CONNECTION ERROR. J = ---

Routine: COFS

Same as 4. Error detected at segment J.

20. ZERO TIME STEPS REQUESTED

Routine: MAIN

Both TMAX and NTSTEP on First data card were zero or blank.

Appendix C

List of Scientific Subroutines Used in TIMDOM

ABS(X)	=	absolute value of X
AIMAG(Z)	=	imaginary part of the complex number Z, (result is real)
AINT(X)	=	integer truncation (result is real)
ALOG(X)	=	the natural log of X
ALOG10(X)	=	the log to the base ten of X
ASIN(X)*	=	arcsine of X, result in radians
ATAN2(X_1, X_2)	=	arctangent of X_1/X_2, result in radians covering all four quadrants.
CEXP(Z)	=	complex exponential (e^Z)
CMPLX (X_1, X_2)	=	formation of a complex number ($Z = X_1 + j X_2$)
CONJG (Z)	=	conjugate of the complex number Z
COS (X)	=	cosine of X
DABS (X)	=	absolute value of a double precision number (result is double precision)
DATAN (X)	=	arctangent of a double precision number (result is double precision)
DLOG (X)	=	natural log of a double precision number (result is double precision)
DSQRT (X)	=	square root of a double precision number (result is double precision)
EXP (X)	=	exponential function (e^x)
FLOAT (K)	=	real number equivalent of integer K
IABS (I)	=	absolute value of integer I
REAL (Z)	=	real part of the complex number Z
SIGN (X_1, X_2)	=	sign of X_2 times $\|X\|$.
SIN (X)	=	sine of X
SQRT (X)	=	square root of X

* Note, this routine is not found in the Honeywell/GE scientific library; therefore, it is included as a subroutine in the TIMDOM deck for this machine.

METRIC SYSTEM

BASE UNITS:

Quantity	Unit	SI Symbol	Formula
length	metre	m	...
mass	kilogram	kg	...
time	second	s	...
electric current	ampere	A	...
thermodynamic temperature	kelvin	K	...
amount of substance	mole	mol	...
luminous intensity	candela	cd	...

SUPPLEMENTARY UNITS:

plane angle	radian	rad	...
solid angle	steradian	sr	...

DERIVED UNITS:

Acceleration	metre per second squared	...	m/s
activity (of a radioactive source)	disintegration per second	...	(disintegration)/s
angular acceleration	radian per second squared	...	rad/s
angular velocity	radian per second	...	rad/s
area	square metre	...	m
density	kilogram per cubic metre	...	kg/m
electric capacitance	farad	F	A·s/V
electrical conductance	siemens	S	A/V
electric field strength	volt per metre	...	V/m
electric inductance	henry	H	V·s/A
electric potential difference	volt	V	W/A
electric resistance	ohm		V/A
electromotive force	volt	V	W/A
energy	joule	J	N·m
entropy	joule per kelvin	...	J/K
force	newton	N	kg·m/s
frequency	hertz	Hz	(cycle)/s
illuminance	lux	lx	lm/m
luminance	candela per square metre	...	cd/m
luminous flux	lumen	lm	cd·sr
magnetic field strength	ampere per metre	...	A/m
magnetic flux	weber	Wb	V·s
magnetic flux density	tesla	T	Wb/m
magnetomotive force	ampere	A	...
power	watt	W	J/s
pressure	pascal	Pa	N/m
quantity of electricity	coulomb	C	A·s
quantity of heat	joule	J	N·m
radiant intensity	watt per steradian	...	W/sr
specific heat	joule per kilogram-kelvin	...	J/kg·K
stress	pascal	Pa	N/m
thermal conductivity	watt per metre-kelvin	...	W/m·K
velocity	metre per second	...	m/s
viscosity, dynamic	pascal-second	...	Pa·s
viscosity, kinematic	square metre per second	...	m/s
voltage	volt	V	W/A
volume	cubic metre	...	m
wavenumber	reciprocal metre	...	(wave)/m
work	joule	J	N·m

SI PREFIXES:

Multiplication Factors	Prefix	SI Symbol
$1\ 000\ 000\ 000\ 000 = 10^{12}$	tera	T
$1\ 000\ 000\ 000 = 10^{9}$	giga	G
$1\ 000\ 000 = 10^{6}$	mega	M
$1\ 000 = 10^{3}$	kilo	k
$100 = 10^{2}$	hecto*	h
$10 = 10^{1}$	deka*	da
$0.1 = 10^{-1}$	deci*	d
$0.01 = 10^{-2}$	centi*	c
$0.001 = 10^{-3}$	milli	m
$0.000\ 001 = 10^{-6}$	micro	μ
$0.000\ 000\ 001 = 10^{-9}$	nano	n
$0.000\ 000\ 000\ 001 = 10^{-12}$	pico	p
$0.000\ 000\ 000\ 000\ 001 = 10^{-15}$	femto	f
$0.000\ 000\ 000\ 000\ 000\ 001 = 10^{-18}$	atto	a

* To be avoided where possible.